HyperTEXT ‘87

Comdex is a Rorschach blot: What you see is what you want to see. What we saw was text: text management software, such as Lotus’ Agenda, Persoft’s IZE, NETI’s DocuForum; text (and image) scanners, such as Palantir’s line and Datacopy’s OCR 4mat; text-based e-mail and groupware systems, such as Lifetree’s Net Results and Conetics’ Higgins. Comdex convinced us -- and another conference (HyperTEXT ‘87) the following week confirmed -- that the big advances over the next few years won’t come from better spreadsheets or even from prettier output, whether of text or data (e.g. database publishing by Ashton-Tate’s Byline and spreadsheet publishing by Microsoft’s Excel). The excitement will come from a profusion of tools to manipulate text.

Until now it’s been tougher to automate the meaningful manipulation of text than of numbers, and the industry has hardly even tried. So far we’ve dealt with text mostly the way desktop publishing treats numbers: It can print them out and make them look good. Now we’re about to deal with text the way spreadsheets deal with numbers -- totals, calculations, sequences, averages. Better yet, we may even reach the text equivalent of graphs and charts, with hierarchies/tables of contents, summaries, and other abstractions.

Why text? Why now?

The business of manipulating text is just starting, and HyperTEXT ‘87 was its christening. Baby products are sprouting up all over -- OWL’s Guide, Apple’s HyperCard, Lotus’ Agenda (and parent Xerox’s NoteCards). As more text shows up on-line, users are demanding tools to deal with it. And the availability of the tools is fostering the creation of on-line text.

The payoffs will be huge. Most of us deal with text far more than with numbers. How excited would you have been to read an article starting like this: \((y-z)\sqrt{56.06} = 45c_{3n}/b+a.\)
\[345 + 4y = 45x...?\]

However, within a few years "hypertext" -- the creation and representation of links between discrete pieces of data -- will be subsumed within the field of text (not word) processing, just as calculations and numerical manipulations are considered part of data processing, and string matches and sorts part of database management. (Never mind that all these functions occur in most applications). Text will have its

FORUM DETAILS ON PAGE 15

EDventure Holdings Inc., 375 Park Avenue, New York, NY 10152, (212) 758-3434
own equivalent of relational views\textsuperscript{1}, called hierarchies or webs or "contexts:" groups of text items filtered by a user arbitrarily selecting nodes or links, or generated automatically depending on specified attributes of the links or text items. Further, one can automatically refilter and recombine nodes. For example, you could display all items -- attendees and topics addressed -- at a meeting, then select one person and display all the tasks that person oversees, then display all the people charged with one of those tasks who have fewer than two other tasks.

But let's begin at the beginning (from the top of the hierarchy, that is)...

**What is hypertext?**

In fact, we'd rather not try to define it. That leads only to academic arguments (of which there were several in Chapel Hill) and adds little to the evaluation of any particular piece of software (just as classification as "AI" is irrelevant). Of course, much of the confusion in explaining hypertext is that there are no agreed-upon definitions, let alone implementations. (Tektronix uses the term "context," for example, specifically in reference to time-oriented versions, while other groups use it to refer to any arbitrary selection of nodes; Agenda uses "views.")

In general hypertext builds upon (not is) the notion of linking pieces (nodes) of information, allowing a user to "navigate" at will through a body of discrete information chunks. There is some information content in each node itself, and more is added by its link(s) to other nodes. That added value can be the reduced cost of finding a node, the intellectual effort of determining that two nodes are related, or both. Sometimes the relationship is simple (an attribute match, for example); sometimes it must be explained (an analogy, a consequence or example of a general statement, etc.).

**Link but don't bind**

Hypertext is as yet a vague notion which has many implementations. Moreover, it need not be standalone: Like expert system rules, hypertext links can enrich many kinds of applications. HyperCard is a file manager and a rich object-oriented graphics environment enhanced with links and a programming language; Agenda is a clever relational dbms enhanced with pattern-matching/text search and a powerful language for manipulating its own data stuctures; SunSimplify (Release 1.0, 31 October 1986) is a rich graphics-oriented relational dbms that represents links in its tables; Guide is a word-processor enhanced with links; KnowledgePro is an expert system shell enhanced with links. Links can point to other locations in a database (or outside it), or they can point to code which is executed (cf. a GOTO statement). In other words, hypertext need not be limited to a specified set of functions but can be enriched or enrich other systems in many ways.

Hypertext can store text, bit-maps or vector graphics, or provide links to VCRs and other equipment. Hypertext can represent highly structured information (part of which is represented by the values of the links themselves).

\textsuperscript{1}That is, a view shows items from separate tables (relations) temporarily joined into a single table, such as a list of customer names and orders from one table and their addresses from another, matched by common values in the "Name" and "Customer" columns of the respective tables.

Release 1.0

25 November 1987
such as decision trees, hierarchies (object-oriented and otherwise), and compositions (where A includes B and C). Indexes -- structured information linked to unstructured information -- are also a form of hypertext.

HyperTEXT '87

HyperTEXT '87 was the first general gathering of the hypertext clan, sponsored by IEEE's Computer Society, Association for Computer Machinery, and the University of North Carolina. Co-chairmen John Smith (UNC) and Frank Halasz (MCC) had originally expected about 100 people. In the end, almost 200 came, and more than that were turned away. Notably, at this conference the luminaries -- Brown's Andy van Dam, Doug Engelbart, Ted Nelson, Fred Brooks -- came at the beginning and stayed through till the end.

The attendees comprised a fertile mix of classicists, computer scientists, professors and business types, and a wide range of attitudes. The hypertext community divides across a number of lines -- first, those who see it as a public medium for publishing and communication, and those who see it as a personal tool. There are those who see it primarily as a way to represent and display human thought, as laid out by a human, and those who see it as a tool or application to manipulate information. And finally, there are those who see it as hypertext, and those who prefer the term hypermedia, to include graphics, audio, video and other forms.

Some were glad that everyone got a chance to see HyperCard, the package that has blessed this field (more by virtue of its name than of its content). Others considered the attention and the platform given to Apple to be an unforgivable sell-out. We figure even the latter will have benefited from the chance to know their enemy. Certainly the HyperCard demo and those in the foyer added a lot to the event.

With a small group, there was a lot of interaction, and also a rueful sense that this was the last time any hypertext gathering will be of manageable size. We could already look forward to looking backward... Remember when 200 of us gathered in Chapel Hill, and there were no standards, and we had to keep explaining things, and everyone used different words for everything? Remember when applications were all tied up inside the hypertext? Remember when Ted Nelson first showed Xanadu to a large group? Remember when Eric Drexler was just one more face in the crowd?

The full title of the conference was "HyperTEXT '87: Systems, applications, issues." Papers on systems (and prototypes) were more plentiful than those on applications, but an encouraging number of people are doing real work in hypertext. Our favorite user talk was by Darrell Raymond and Frank Tompa of the University of Waterloo, on putting the Oxford English Dictionary onto CD-ROM -- a talk rich in practical experience and love for words. The "issues" as yet were mostly technical, but just wait: Who gave you the right to link me to your ridiculous advertising? as several revered ancients might ask.
Hypertext publishing: The whole wired world

Long-term, as discussed but not resolved in North Carolina, the notion of hypertext as an open publishing medium implies the ability not just to transfer files but to establish permanent links across systems, so that any person can read and append his comments to any text "published" anywhere in the worldwide hypertext. (The major proponents of hypertext publishing are Ted Nelson and Eric Drexler, currently a visiting scientist at Stanford.) This approach is not yet broadly implemented, nor was it formally addressed.

Dynamic text

In its basic form, hypertext is static, just as a database is static. It takes an application to create and manipulate the database -- or the hypertext. (See the table on page 7 comparing hypertext, relational dbms, and object-oriented databases; also, Release 1.0, 10 August and 21 September.) It represents that intellectual effort but doesn't perform it. It lets you follow links. However, with the exception of Tektronix's HAM (page 10), most current hypertext systems mingle a static hypertext base with an application, much as applications used to contain both data structures and code.

Much interesting discussion at HyperTEXT '87 centered on the creation of applications and facilities for creating, querying and manipulating hypertext bases. Searching on and defining views by content is useful (and is extended nicely in Lotus Agenda with its ability to include fields/categories in its text items), but adding value to the links themselves and defining and parsing the link structure can add even more power. When the links get truly defined and regular, you end up with some more traditional data structure that is easy to manipulate in more traditional ways.

Typed links/filters

Links have three basic roles: pointers for navigation (by a person or application), representations of relationships, and typed links for filters to act upon. (In fact, you could represent links as a special kind of node, or list them in a relational database in tables of "typed pointers." ) Links can trigger many kinds of actions, such as moving to the new location (the most common), inserting the item or replacing the original item with the one linked to, as in Guide. They could also represent is-a-part-of, is-a-kind-of or is-an-instance-of (as in an object-oriented database). Yet other links could be "supports an argument," "follows from," "is a citation for," "there is a flight from X to Y," and the reverse of all these. These links could also have values: certainty factors, equations, flight times, distances, that could be queried, reasoned about, calculated or otherwise manipulated. The values of various paths through a hypertext base could be computed, or links could be traversed as long as certain conditions were met, and the system could display just the endpoint, or some or all of the nodes traversed along the way.

It's that arbitrary structure that gives hypertext its unique nature -- and that makes it uniquely difficult to reason about. It is the resolution or abstraction of a seemingly arbitrary structure into a more regular structure that can be manipulated, queried and reasoned about that is the work of an application or a person, and that ultimately adds value.
Queries and applications

As yet, there is no general language for manipulating hypertext, and it will be hard to build one given the multiplicity of possible link types. Yet there should at least be a general way of defining and finding certain structures (A query if not an application language): Find an item with at least four support links with certainty factors greater than 80 percent. Another favorite is to find all the circular arguments; that is, arguments where A supports B supports C supports D supports A, with an arbitrary. Or find all pictures that have no captions or text references. Find all the counties that include cities served by more than three television stations.

Or from a hypertext base of flights between cities, show all routes between New York and Aspen that don't go through Chicago and take no more than two stops. (Add to that a rule-based system for determining fares...) Unless you had laboriously entered all connecting flights, that would be virtually impossible to do in a standard database, but quite easy with the right kind of hypertext tools (as suggested by Alberto Mendelzon of the University of Toronto). Hypertext could also represent a program -- either a messy one with lots of gotos, a series of hierarchies (structured), linear (straight-forward), or a decision tree or expert system (a regular hierarchy). If you look at or edit the program, it's hypertext, with a context-sensitive editor/application that displays program flow; if you execute it, it's the application itself. (Call these metaviews.)

We suspect much of the expertise to build hypertext structure editors will come from the CAD/CAM and CASE communities, accustomed as they are to dealing with visual imagery and perspective, spatial relationships, etc...

Hypertext as knowledge representation: Abstraction

Now let's consider hypertext as a means of representing text. Aside from hypertext publishing (page 6), it's about the journey from random facts to some kind of structure and -- at least in its use as a writing tool as described by John Smith of UNC -- into a linear form for transmission into someone else's mind, with its own rich associative memory. Then it may go back into that other person's own hypertext system.

Is this abstraction necessary? Is the effort involved in placing all these random thoughts into linear order an unfortunate, unnecessary cost of the medium in which we work (and you read), or is it a necessary part of the process of understanding and clarifying these thoughts? Certainly the creation of the links is necessary to full understanding; perhaps their formulation into a linear progression or even a strict hierarchy is not. Traditionally, linearization has been required simply to transform ideas into written text (or speeches and other formal presentations). Indeed, even interactive hypertext sessions are linear, but the particular sequence is under the user's control.

However, we would argue that abstraction if not linearization is still a necessary part of the process of passing information into another person's mind for mapping onto that second person's information structure. It's difficult to communicate a massive amount of information without meaning -- or the abstraction that represents meaning. It's in seeing the connections that people learn, that things make sense enough to remember, etc. They look for the structure and the framework of the information they need, re-
membering the generalities and where to look for the particulars. What is meaning but generalizations, theories, analogies, that ultimately enable one to classify and thereby predict? For example: "Here's an argument about abstraction and details, and here are some examples." "Here's a plot and here are the details." "Here are some experiments and the theory they support." "Here's a map of France with its top 10 cities." The examples, the details, the experiments, the locations, are all meaningless and unmemorable (unselectable?) without some structure to hang from. (Remember that pejorative term "rote learning")

Filtered views or arbitrary alternative "webs" allows the automatic flexibility to display ideas in different structures, the text equivalent of seeing a four- or five-dimensional object from different perspectives. Yes, the point is that you may never see the "whole" thing.

As we write this essay, with thoughts floating around in our mind, we're well aware of this process: First, we type in some random thoughts. Next we set up a number of headings above those random thoughts, and proceeded to write yet more things under them. We move the headings around, with their associated text. We move blocks of text from one place to another. We reorder the sentences within a paragraph. Each sentence leads to several other thoughts: Which should come next? And how can we return to the second thought gracefully? Should object-oriented databases be mentioned in the main text, or relegated to the table caption on page 7?

Our selection of the order in which ideas are presented is in part forcing our way of thinking onto others. But if it is well done, it may be a valuable contribution to rendering those same ideas intelligible. If what you learn is what you already almost know, then the sequence in an argument can be structured to get you to that point: Some paths are better than others.

Besides, the clumsiness of paper cross-references is a gentle, useful restraint that forces writers to offer up only the most valid of links -- a discipline just as lack of space is a discipline for programmers or artists. Of course, the discipline of paper does not always provoke the appropriate response. We have read many paper documents that have an order, but no discernible logical links between their parts.

Standards and diversity

A hot topic at the conference was the notion of standards. These come in three parts: user interface and conventions, file (node) structures, and link protocols/behavior. The case for user interface standards is simple, except that no one can agree what they should be. The case for file standards for interchange of hypertext data is likewise easy, and should be possible to implement for transfer of the nodes' data. However, the standard for links should and can only be minimal, providing basic ways to describe links and their behavior, without limiting what an application can build on top of that. A group of vendors is working on this issue, linked and coordinated by Amy Pearl of Sun Microsystems and Norm Meyrowitz of IRIS.

It's likely that various developers will slowly adopt the formats of -- for starters -- HyperCard on the Mac, Guide and Agenda on the PC, and Knowledge Management Systems on the Sun, and eventually standards will coalesce. In fact, it's a little early for standards; the facilities of all the products named are still limited and limiting.

Release 1.0

25 November 1987
THREE DATA STRUCTURES

<table>
<thead>
<tr>
<th></th>
<th>hypertext</th>
<th>relational dbms</th>
<th>object dbms</th>
</tr>
</thead>
<tbody>
<tr>
<td>data types</td>
<td>anything, in nodes and links</td>
<td>values, strings (alpha sequences)</td>
<td>anything, incl. procedures</td>
</tr>
<tr>
<td>links</td>
<td>explicit, random</td>
<td>implicit only</td>
<td>structured</td>
</tr>
<tr>
<td>types</td>
<td>arbitrary types</td>
<td>not typed</td>
<td>limited types</td>
</tr>
<tr>
<td>creation</td>
<td>arbitrary, static</td>
<td>based on matches, dynamic</td>
<td>based on structure</td>
</tr>
<tr>
<td>typical search</td>
<td>by location</td>
<td>by value</td>
<td>by structure</td>
</tr>
<tr>
<td>structure</td>
<td>arbitrary graph structure</td>
<td>flat, tables</td>
<td>hierarchies</td>
</tr>
<tr>
<td>language</td>
<td>Agenda C&amp;A, HAM operators</td>
<td>SQL</td>
<td>C++, Objective C, Smalltalk</td>
</tr>
<tr>
<td>typical applications*</td>
<td>text storage, manipulation</td>
<td>transactions, data analysis</td>
<td>modeling, managing rich data</td>
</tr>
</tbody>
</table>

*Adherents of each approach consider it suitable for everything.

The table above attempts to clarify the commonalities and differences among three approaches to logical data structure. Of course, any data structure can be represented as hypertext, but if you have a regular structure, you get far more power and manipulability by using it explicitly (just as you get better control of text in a table by using tabs rather than the space bar). Object-oriented databases are simply a special case of hypertext systems, with highly typed and regular links, plus inheritance and a few other things, or you could say that hypertext is a special case of oodb, with links to other objects in each object's "slots." Underneath, of course, the information in any of these logical forms is stored on some physical medium (optimized in the case of object-oriented or relational dbmses, which can be used for physical storage of hypertext); here we're addressing how access to it is organized and how it is presented to the user or to an application.

A hypertext structure does not exist independent of the data within it. A relational database has flat tables and can derive further implicit structure that is derived when an application or query requests a join. (A knowledge base, incidentally, is a more abstract construct that may consist of rules, objects (or frames), procedures, and anything else that represents "knowledge." It's more a way of looking at information than of organizing it.) Like a dbms, hypertext will ultimately take form as a separated front end and back end: The back end provides facilities and holds data structures, while the front end holds the less generic data structures, a particular interface, display conventions, etc. (An object-oriented database puts as much of such information as possible back into the database.) This separation lets different applications be optimized for different tasks and user classes. Maybe one day we'll have an SQL (or PostScript) of hypertext. For the moment, we are in the stage where data, data structures and applications are inextricably mixed. We still need to experiment before we precipitate them out.

Release 1.0

25 November 1987
Implementation issues

With a few exceptions, most hypertext systems are limited either in function or robustness (or not yet implemented). Issues of concurrent use and version management (or concurrency across time) are challenging and will increase in urgency as hypertext use broadens. What happens to links when the nodes are split, moved, deleted, or changed in any way (cf. the question of what happens to spreadsheet formulas when changes are made elsewhere in a worksheet)? For example, when node A1 is changed to become A2, does B1's link to A1 now link to A2? Or does it still link back to A1? There is no general answer, but a default must be specified for each system, and its consequences dealt with.

How do you manage conflicting versions of links and reconcile various users' changes? How fine can the granularity get? How can you restrict access appropriately when anything may be connected to anything? These questions are tougher the less structured your data is. In general, the answer will be optimistic concurrency control; that is, anyone is allowed to modify the data, and reconciliation of the changes occurs when the changes are stored. The first user to record his changes wins, unless it is specified to the contrary. Later users must then do their work based on those changes. (A well-designed system will notify the user that someone has also checked out the data he is working on, and will further notify him when another user's changes are registered.)

Particularly hard questions arise when you want a node to belong to two compositions, or to have parents in separate hierarchies (multiple inheritance). As Fred (Mythical Man-Month) Brooks put it in a superb dinner speech, "Sooner or later you come across a duck-billed platypus."

Intellectual property issues

At the end of the conference, technology gave way to issues of content and ownership. Mark Bernstein of Eastgate volunteered to publish a digest of the proceedings as hypertext (in his own Hypergate\textsuperscript{TM} format), with Bernstein's omissions and links but no changes to each person's textual contributions. Jef Raskin of Information Appliance asked if he would have a chance to review the omission made from his work, another person questioned the addition of "spurious" links, someone else asked about copyright, and suddenly an academic discussion became real. These are all issues, to be sure, but why does technology suddenly turn people into such rigid thinkers? Most doctors make mistakes far more frequently than an expert system. Most human editors make spurious "links" and even changes to authors' texts. The mere presence of technology should not require the presence of a review board. (And who shall review the review board?)

The issues of ownership and copyright will indeed be challenging. If worldwide publishing happens, how do authors get paid, and what control do they have over their work? How do they protect it from "spurious" links? Who owns the copyright to a restructured document? Certainly the original owner has some interest in it, but what of the intellectual effort that went into the linking? As hypertext blossoms, it will become clear that at least some links contain intellectual effort that is deserving of protection. Can an author protect his work from others' links? Is the traversal of a link a service that should be paid for by the "reader"?
INSTANCES OF HYPERTEXT

The set of hypertext systems beyond prototype stage is still fairly small. Perhaps the most broadly used (except for HyperCard and OWL's Guide) is Xerox's NoteCards. Originally a personal tool, NoteCards is now generally used in collaborative work. It is a straightforward hypertext system, with a rich visual display, limited typed links, and the notion of FileBoxes, or a hierarchical system for complex NoteCards networks. Other broadly used systems include InterMedia, the best developed implementation of the concept of hypertext including media besides text, developed at Brown University by the Institute for Research in Information and Scholarship (IRIS). About 20 people, mostly at IRIS, use it regularly to do their work, and some 70 students in two classes have used it for work in English literature, biology, and religion. At the University of Southern California, a hotbed of software engineering research, Walt Scacchi and others are building a hypertext system to manage software documentation and code modules. Some other notable systems, both working and rudimentary, are described below.

gIBIS: People and hypertext

When used as rich multi-user tools for argument/discussion, hypertext systems tend to impose the same kind of explicitness as certain e-mail systems do on interpersonal communications (Release 1.0, 24 September 1986). An interesting example of this is gIBIS (graphical Information Based Information System), an MGC system built on top of SunSimplify and used to represent multi-person "reasoned discourse" -- that is, polite discussion of views and counterviews. Link types include supports, responds-to, is-suggested-by, and generalizes, while nodes include issues, arguments, and positions. One gIBIS user told designers Jeff Conklin and Michael Begeman that it "exposes axe-grinding, hand-waving, and clever rhetoric." We suspect that some people will feel uncomfortable with such a system much as some do with Action Technologies' Coordinator's tendency to remind people explicitly of what they have promised...but clear thinkers will love it! gIBIS, like The Coordinator, requires people to be explicit: "Is that an idle remark, or a request?" "Do you intend to support this statement, or to refute it?" "Which side are you on?"

EUCLID: Communication and display

The public (or that small portion that has heard of the term) generally considers hypertext as something visually rich like NoteCards, HyperCard or Guide, rather than visually austere, like Agenda (call it sleek, to be polite). But these systems only reflect back what was put into them. One of the more interesting systems discussed, built by several people at UC Boulder, uses a constraint-based display generator to position text items on a screen according to the types of links that connect them. The sample application, EUCLID, lines up contradicting positions on either side of a screen, with supporting statements arranged beneath them in a way that's immediately meaningful to human viewer. This facility could make it easy to transfer rich information across a "thin" communications medium, regardless of the particulars of the display on either side. Other implementations of the meaning-defines-display concept could do the same for a time-oriented system, or one about project-planning, etc. (Of course, many calendar applications already do this, but in an application-specific way.)
HAM: The Hypertext Abstract Machine from Tektronix

How many stepladders does it take to reach the moon? Can hypertext really "work" when it's applied to serious problems? In fact, like most automated information systems, hypertext applications should come into their own when they're applied on a large scale. That's when the arbitrary structure of hypertext is likely to break down, and the need for organizing structure and manipulation tools will become apparent.

Perhaps furthest along in the use of hypertext for applications rather than information presentation is Tektronix, with its Hypertext Abstract Machine (HAM). HAM is based on the underlying system for Neptune, a prototype application for program development written in Smalltalk by Norm Delisle and Mayer Schwartz of Tektronix Laboratories in 1985. It now also supports Tektronix's Software Engineering Information System, a more robust, production-style environment than Neptune designed ultimately to support Tek's suite of CASE tools. HAM is used regularly by six people, including its designers.

A flexible transaction-oriented, multi-user system, HAM manages multiple versions of nodes in "contexts" (defined across time) and mediates among different users. It also includes rich facilities for typing both links and nodes, and can execute queries both by node or link attributes and structures. Brad Campbell and Joseph Goodman showed how HAM could be configured to emulate the performance of Guide buttons, InterMedia webs, and NoteCards FileBoxes. Guide buttons come in several varieties (insert, replace, goto); NoteCards FileBoxes are essentially virtual collections of NoteCards nodes that can be manipulated as one item; and InterMedia webs are alternative sets of links among hypertext nodes, designed to foster a user's exploration through text according to some particular view of the information presented.

The Xanadu Project

Ted Nelson, the father of this long-in-gestation field, made a heartfelt and funny speech and brought along a prototype of Xanadu. Xanadu is a global medium for text storage and hypertext publishing, and was built with the understanding that third parties would build interfaces. Designer Ted Nelson coined the word hypertext in 1964, and is just now, at the age of 50, seeing his ideas gain widespread use and acceptance. His book Computer Lib has just been re-released by Microsoft Press, and he has himself re-issued Literary Machines Version 87.1. (Nelson deserves more space than he gets here, and he will get it now that hypertext is fashionable.)

Lotus Agenda

More practical than Apple, Lotus is not positioning Agenda as a hypertext product. Unlike typical hypertext interfaces, in Agenda most links aren't explicitly built as links, but rather are derived from field values, by where the user types information on the screen or by the use of keywords, singly or in patterns. We consider the product an excellent example of appropriate integration of hypertext and other facilities, notably relational data management. Moreover, Agenda's "conditions and actions" language is the closest thing we've seen to a hypertext language, one that manipulates data and structures. By contrast, HyperCard's HyperTalk is not about the HyperCard data structure, but is a more traditional programming language, albeit a friendly one.

Release 1.0 25 November 1987
MITCH MOVES ONWARD: EXTENDING THE OS

In the past couple of years, the notion of what belongs in an operating system has broadened to include database management, interprocess and cross-machine communications, and facilities such as HyperCard's. ON Technology's goal is to extend it with a variety of specific capabilities: support of groupware, object-oriented structures and languages for handling rich data and real-world knowledge, text management, usable interfaces. ON Technology has a good chance of succeeding both because of and despite its great ambitions. To be sure, ON is not alone in its goals, and in the next few issues we will discuss many companies who share all or part of its ideas.

After a year of searching, Mitch Kapor has fallen in love again, with something big enough to build a company around. President to his chairman at ON Technology is Peter Miller, formerly with Lotus and DEC and an expert both in AI and compiler/editors. They don't necessarily want a large, "bloated" company, says Miller, but "absolute control of the things we care about." Their aim is to provide a platform for the next generation of workstation applications, fitting in between the operating system and the applications themselves. "We want to build on what's out there, those millions and tens of millions of pcs and operating systems," says Kapor, rather than foolishly attempt to replace the Mac OS and OS/2. With luck, ON will show up before the next generation of applications is fully established -- and may help make it happen. Distribution channels are still unclear, but Miller and Kapor will be on the hustings over the next couple of years trying to win ISVs and OEMs to their side. Kapor and Miller should at least get a fair hearing: Many people feel that the man who designed 1-2-3 can't be all bad, regardless of how they feel about the company now selling it.

Rather than just a cunning glue to hook together existing applications, but ON's platform will be something more fundamental on which better applications can be built. This, by the way, does not mean the erstwhile "integrated software," but a supporting environment that can handle the same data in many ways. It's like a spreadsheet that can handle text and numbers, but more so. An underlying facility will be some kind of object-oriented database, capable of handling documents, messages, data, whatever users can throw at it. (See Release 1.0, 24 March and 10 August.)

Aside from data representation tools, ON's platform will include models for typical discrete objects in the world -- people, companies, locations, etc. -- just as a calendar manager knows about time or a typical accounting application knows about profits, taxes, and divisions. Why not abstract out these common notions and represent them generically? All employees contribute to payroll costs, even though compensation details vary. The trick is to express the relationships properly, but leave the details up to the application-builder -- doing much of the work that would be done in a canned application, but still leaving maximum flexibility.

Miller and Kapor have made the long journey through AI and beyond. AI is enticing, but it's only a small part of the answer. Its promises -- codification of expertise and policy, representation of rich knowledge, flexibility, "friendliness" -- are best delivered by a combination of techniques including rules, object-oriented database management and language, hypertext, and graphics. Mitch Kapor's track record of achievement is enhanced by his clear vision of his own limitations -- and his pragmatism. Says he, "If Alan Kay is Christopher Columbus, we want to be the East India Trading Co."

Release 1.0 25 November 1987
"This bank does not pay me to build prototypes," says Craig Atkinson. "It pays me to build applications. The prototype is just a stage. We're not trying to prove a point; we're trying to seed this technology throughout the bank." Atkinson, vp of advanced technologies for the information and technology management division of Chemical Bank, is a practical guy (like most people who are successful with expert systems), and loves to quote his boss: "AI is not a religious experience; it's an economic experience."

In keeping with this spirit, Atkinson and a team of four people formed a partnership with Charlie Caputo, Chemical's vp of edp auditing (the man with the R&D funds). Together, they took just four months to build a working, useful expert system after the delivery of a Symbolics and Inference Corp.'s Automated Reasoning Tool (ART) last November. That system, FX Auditing Assistant, is typical of the successful expert systems we've seen: It's not all that complicated, and it doesn't insult the intelligence of the user. It says, in effect, "I'll do the boring part, and you do the clever part." As far as the "boring" part goes, the system is going to be as reliable as a conscientious human could be given all the time he needed -- and it leaves the human time to concentrate on the clever part.

The wonderful world of foreign exchange

Given trading's contribution to Chemical's bottom line -- substantial but unrevealed -- and the increasing volatility of the foreign exchange (and all other) markets, foreign exchange seemed ripe for some kind of attention when Atkinson started to look around for ways to apply expert systems technology in mid-1986. Chemical Bank does roughly $750 billion per year worldwide in foreign exchange and is one of the top 10 in the business, generating thousands of transactions each day. Each transaction has a date, time, price, amount, profit or loss, buy or sell, currency, broker, contraparty (the other side of the deal), and a branch office (typically New York or London). With the date for each transaction taking one line of printout, a month's worth of trades can easily weigh 10 pounds.2 Imagine trying to audit that! Before this system, auditors would go through all those pounds of paper in search of suspect trades. With FXAA, they are now picked out and displayed in a couple of ounces.

"Are we trading properly?" is the question. It's a tough one to answer. The final factor, of course, is results, and lack of legal problems, but there's a lot that goes into that. Although every trade is a judgment call, the first measure is to compare trades against market conditions -- i.e., prevailing prices. What's a prevailing price in an ever-changing market? Chemical uses two measures based on its own transaction prices each day: average and closing prices. In the foreign exchange world, of course, even that isn't as simple as trading your dollars at a discount at an airport bank.

About 80 percent of foreign exchange trades are in the spot, or immediate-delivery (two days actually), but about 20 percent are in fact futures con-

---

2Dave Liddle, chairman of Metaphor Computer Systems, a high-end analysis workstation now selling mostly to marketing types, once told us how he found customers: "Look for the printouts." Foreign exchange has them!
Here's a rough sketch of the price curves and dots representing trades on the FX Auditing Assistant screen. Because the system uses live data, it was not possible to get an actual screen shot.

tracts -- for delivery anywhere from a few days to a few years into the future. Those prices depend on prevailing expectations for exchange rates, and form a relatively smooth curve when you plot prices against delivery dates. In the end, Chemical took two approaches.

On screen

FX Auditing Assistant comes in two parts, suited respectively for people who use screens and like interactivity, or for more traditional people (such as many auditors) who really prefer the tangibility and familiarity of paper. The screen version displays a plot of each day's transactions (little dots) against two curves -- average and closing prices. It's easy to see which trades are "off-market," and the user can select them with a mouse to display the particulars of the transaction on the lower part of the screen.

Now the fun begins: Why was this transaction off-market? Most frequently, it was a retail trade -- a few thousand dollars, with a high variance, usually favorable to the bank but relatively small in absolute dollars. (We can imagine a refinement of the system that would display certain trades differently -- by color, or as crosses instead of dots.) Such transactions can be dismissed quickly. The next thing you might look for is to determine if a trade was part of a swap -- where there's a roughly offsetting transaction with the same party. Ask for all the transactions with the same party, and the answer is easy to determine. Or it might have been a pass-through, where the amounts of two trades cancel exactly, and the same broker shows up on both sides. In this case, there's no net P&L, but there may be a transaction cost (payable to the broker) that's not reflected here. Is a certain broker getting too much of such business? You can inspect all the trades with that broker.

As you use the system, you always have the option of adding any transaction to a list called "interesting transactions." You can then examine this file, which may include transactions from many different days, at your leisure. (Price variances are always determined in relation to the price curve on the day the transaction occurred, but relationships with third parties persist across long periods.)

On paper

While the screen version looks stunning, it's the paper version that contains the system's real smarts. The screen version lets you select transactions to look at; the paper version selects the transactions you should look at. The system produces a printout much like the traditional one, but something's missing...all the transactions that aren't unusual in any way. The system then lists off-market trades (ones at currency rates greater than a
specified deviation from the average) on a traditional printout, with likely swaps and passthroughs grouped together. The largest part of the list we saw was about eight pages of off-market trades in small amounts -- classified by the system as "probably not interesting." Then there's the list of "interesting" trades that fit no known classification -- potentially the most interesting of all. Now the human can deal with a list of trades small enough to grasp -- and perhaps find something common among those trades. Sometimes it can be as trivial as a bad day; in a number of cases, it turns out to be a data-entry error -- reversed digits or a dropped zero. But on occasion it may signal something worth investigating.

Could FXAA it ever miss a "bad" trade? Sure, but not a suspect one, according to the rules that a trained auditor would use. If the human auditor begins to harbor a certain suspicion, he can change quickly the rules FXAA uses to flag suspect trades. "Suppose we open the paper one day and see that another bank has had huge trading losses in a certain currency, or with a certain broker," says Atkinson. "It's easy for us to modify the rules and run our last few months of trading through the system again."

Implementation issues

In toto FXAA has about 60 rules, plus a few decision trees to define "pass-throughs," "swaps," and what constitutes a "small" trade. As described, the system may even sound prosaic. But that's the nature of expert systems: They execute explicit rules that sound simple when you list them one by one. The value is that it was far easier to write these rules, and it will be far easier to change them.

In another way, too, the Chemical system is designed to be practical rather than fancy. Aware of American Express's and other groups' problems integrating disparate technologies, Atkinson limited the links between the Symbolics machines and the trading department's System 38s to "sneakernet." Tapes are messengered between Chemical's Wall Street systems people and its uptown trading facilities. "All I have to do is build an expert system on our mainframe and have some [unrelated] glitch make our corporate database go down," he says. "Just see how fast this system will go out the door.

In the next few months, the system will be reimplemented on a VAX, still on top of Inference's ART with LISP extensions for the decision trees and other items. Since this work will be done at Inference's Los Angeles headquarters, Atkinson notes with a grin, his group have to come up with dummy data for a system that up till now has used only real transactions.

For the moment, FXAA is handling trades in just a couple of currencies at the New York branch, but it will soon be expanded to a larger range of currencies at other branches (London and elsewhere). Several other banks and corporations have already approached Chemical about buying the system -- a clear sign of its value. Long run, of course, there's no reason the same principles couldn't be applied to trading in other instruments (although not at Chemical because of legal restrictions on banks' trading activities). The range of possibilities -- instead of a handful of currencies imagine thousands of different stocks, preferreds, bonds, etc., from different issuers -- is greater, but the fundamental issues are more or less the same. What are the unusual patterns? Who keeps making money, and who keeps losing? Add to that who knows whom (and what)...

Release 1.0 25 November 1987
FORUM DETAILS

Forum invitations were mailed to subscribers last week. They may be copied for use by others in your organization (two paid registrations per subscription). If you have not received yours and you would like to attend, please look around carefully and then call us for a copy. (We cannot handle registrations by phone.) We sold out last year and expect to sell out this year, so please hurry! (Take advantage of our early-registration discount, tailored for small companies who can cut checks quickly, which expires December 18.)

The Forum will take place February 21 to 24 in Naples, on the west coast of Florida (so that the sun sets over the water). The theme is "Worlds in Collision: From PC to Workstation." We will explore the broadening definition of "personal computing" as users grow accustomed to more and more power on their desktops -- and to closer and closer links to others' desktops. What will be the impact on and of new operating systems, applications, distribution strategies, and shifting balances of power?

Confirmed speakers include:

Victor Alhadeff
Bob Berland
Gordon A. Campbell
Rod Canion
Vittorio Cassoni
Peter Coffee
Bob Epstein
Edward M. Esber
Gordon Eubanks
Robert Flast
William Gates
Phillippe Kahn
Jerry Kaplan
Mitch Kapor
Bill Krause
Jim Manzi
Mike Maples
Scott McNealy
Bob Orbach
Vern Raburn
John Roach
Morton Rosenthal
John Sculley
Larry Tesler
Edward Tufte
David S. Wagman
Kenneth R. Waters
Joyce Wrenn
Haviland Wright

Egghead Discount Software
IBM Application Systems
Chips & Technologies
Compaq Computer
AT&T
Aerospace Corporation
Sybase
Ashton-Tate
Symantec
American Express
Microsoft Corporation
Borland International
GO Corporation
ON Technology Inc.
3Com Corporation
Lotus Development Corp.
IBM Entry Systems
Sun Microsystems
47th Street Computer
Symantec
Tandy Corporation
Corporate Software
Apple Computer
Apple Computer
Yale University
Softsel Computer Products
ComputerLand
American Airlines
Avalanche Development

In addition, there will be a special panel, "Beyond Numbers," on text-based applications, including e-mail, hypertext, and automated paper-shuffling (process management). In the afternoons you may attend parallel company presentations and demonstrations of products and vaporware by some of the speakers listed above and by others.

Release 1.0

25 November 1987
PHONE NUMBERS

Norm Meyrowitz, IRIS/Brown University, (401) 863-2943, nkm@iris.brown.edu@relay.cs.net
Craig Atkinson, Chemical Bank, (212) 493-4132
Mark Bernstein, Eastgate Systems, (617) 782-9044
Jerry Kaplan, GO Corp., (415) 543-3200
Jef Raskin, Information Appliance, (415) 493-2400
Alex Jacobson, Inference Corp., (213) 417-7997
Frank Halasz, MCC, (512) 338-3648, halasz@mcc.com
Conall Ryan, Ed Belove, Lotus Development, (617) 577-8500
Peter Miller, Mitch Kapor, ON Technology, (617) 225-2545
John Smith, University of North Carolina-Chapel Hill, (919) 962-1792
Walt Scacchi, University of Southern California, (213) 743-7424
Alberto Mendelzon, University of Toronto, (416) 978-2952
Ted Nelson, Project Xanadu, (512) 692-7346, (408) 244-2643
Eric Drexler, Stanford University, (415) 364-8609
Amy Pearl, Sun Microsystems, (415) 691-2840, apearl@sun.com
Joseph Goodman, Brad Campbell, Tektronix, (503) 629-3000, 629-1722, joego@copper.tex.com, bradc.copper.tex.com,
Mayer Schwartz, Tektronix Laboratories, (503) 627-6150

COMING SOON...

- Connectivity: Promises, promises.
- Groupware -- software from start-ups.
- Parallel processing.
- Channels -- Micro and otherwise.
- Nitty-gritty experts: Are they intrinsically friendly?
- Retrofit technology: Gluing applications together.
- And much more...

Release 1.0 is published 12 times a year by EDventure Holdings, 375 Park Ave., New York, NY 10152; (212) 758-3434. It covers the pc, software, CASE, groupware, text management and connectivity markets, and artificial intelligence. Editor & publisher: Esther Dyson; associate publisher: Sylvia Franklin; circulation manager: Hyacinth Frederick; copy chief & consulting editor: William M. Kutik. Copyright 1987, EDventure Holdings Inc. All rights reserved. No material in this publication may be reproduced without written permission; however, we will gladly arrange for reprints or bulk purchases. Subscriptions cost $395 per year, $475 overseas; multiple subscription rates on request.
Release 1.0 Calendar

December 4
AGS at the analysts' - New York City. Sponsored by NYSSA. Contact: Terry Earle or Judy Zatz, (212) 344-8450.

December 7-11

December 9-11
Database management systems for engineering and knowledge-based applications - Santa Clara. Seminar course on object management systems led by Mohammad Ketabchi, with specific assessments of Vbase, GemStone and Iris. Contact Dr. Ketabchi at Santa Clara University, (408) 554-2731.

December 14-16
Expert systems and artificial intelligence symposium - Atlanta. With James Martin, Larry Harris (AI Corp.), Herb Schorr (IBM), and others. Sponsored by Digital Consulting, Inc. Contact: Lisa Mosca, (617) 470-3870.

December 17
Automatic Data at the analysts' - New York City. Sponsored by NYSSA. Contact: Terry Earle or Judy Zatz, (212) 344-8450.

January 11-12
Neural networks - Los Angeles. A commercial assessment of NN applications, with most of the major players, followed by a separate three-day course on the technology. Sponsored by the Institute for International Research. Contact: Russell Webb, (219) 883-1770.

January 15-17
Macworld - San Francisco. Sponsored by Mitch Hall Associates. Contact: Peggy Kilburn, (617) 329-8090.

January 25-28

February 8-10
IFIP conference on computers and law - Santa Monica, CA. Issues that just won't go away: Copyright, contracts, taxation, computer crime, legislative actions. Sponsored by IFIP and Los Angeles County Bar Law and Technology section. Contact: Michael Krieger, (213) 208-2461.

February 17-19

February 21-24
ELEVENTH ANNUAL PERSONAL COMPUTER FORUM - Naples, FL. We moved it in search of variety and better weather. Registration forms have just been mailed to subscribers (two per subscription). For further information, please see page 15 or call Forum director Sylvia Franklin at (212) 758-3434.

February 24-26

Release 1.0
25 November 1987
<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
<th>Location</th>
<th>Sponsor(s)</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>February 25-27</td>
<td>Workshop on technology and cooperative work - Tucson, AZ.</td>
<td>Sponsored by Bell Communications Research and the University of Arizona. Contact: Robert Kraut, (201) 829-4513 or Jolene Galegher, (602) 621-7477.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 1-3</td>
<td>Third international CD ROM conference - Seattle. Sponsored by Microsoft. Contact: Sherrie Eastman, (206) 867-3305.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 7-10</td>
<td>IEEE conference on computer workstations - Santa Clara. Sponsored by IEEE. With Sun's Bill Joy, and sessions on distributed systems, computer-supported cooperative work, and OS/2. Contact: Pat Mantey (408) 429-2158 or Robin Williams, (408) 927-1842.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 8-10</td>
<td>Connect '88 - New York City. Sponsored by Cahners, with Datamation and the Gartner Group. A trade show on connectivity and integration directed at corporate end-users. Contact: Richard Holden, (203) 964-0000.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 14-18</td>
<td>Artificial intelligence applications - San Diego. Sponsored by IEEE. Contact: Richard Greene, (301) 468-3210 (exhibits) or IEEE, 371-0101 (program) or Paul Harmon (415) 861-1660.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 16-23</td>
<td>Hannover Fair CeBIT - Hanover, West Germany. Contact: Donna Peterson Hyland, Hannover Fairs USA, (609) 987-1202.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 20-23</td>
<td>ADAPSO SPRING CONFERENCE - Palm Desert, CA. Software and services vendors at the oasis. Contact: Sheila Wakefield, (703) 522-5055.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>March 27-30</td>
<td>Software Publishers Association spring conference - Berkeley, CA. Contact: Jackie McDonald, (202) 452-1600.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>April 7-10</td>
<td>13th West Coast Computer Faire - San Francisco. Contact: Jason Chudnofsky at Interface Group, (627) 449-6600.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Event Description</td>
<td>Location</td>
<td>Contact Details</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------------</td>
<td>---------------------------------</td>
<td>------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>April 11-14</td>
<td><strong>AIIM show</strong> - Chicago. Information and image management. Sponsored by Association for Information and Image Management. Contact: Sue Wolk or Betty Garrett, (301) 587-8202.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>April 11-15</td>
<td><strong>IEEE Tenth International conference on software engineering</strong> - Singapore. From an international perspective. Sponsored by IEEE and NCB Singapore. Contact: Tan Chin Nam or Lim Swee Say, (65) 772-0200.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>April 19-21</td>
<td><strong>CEPS/Spring '88</strong> - Chicago. Corporate electronic publishing systems. Sponsored by Cahners and InterConsult. Contact: Mike Driscoll, (203) 964-0000, or Paula Wertman, (617) 547-0332.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May 9-12</td>
<td><strong>Comdex Spring</strong> - Atlanta. Peaches and PCs. Contact: Jane Wemyss at the Interface Group, (617) 449-6600.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>June 6-8</td>
<td><strong>Artificial intelligence in electronic publishing</strong> - San Jose. Sponsored by the Graphic Communications Association. Applying AI to design, content, process, etc. Contact: Marion Elledge, (703) 841-8160.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>August 22-26</td>
<td><strong>AAAI-88</strong> - St. Paul, MN. The seventh annual. Sponsored by the American Association for Artificial Intelligence. Contact: Claudia Mazzetti, (415) 328-3123.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>September 26-28</td>
<td>Second conference on computer-supported cooperative work - Portland, OR. Sponsored by ACM. Contact: Suzanne Sylvia, (617) 225-1860.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>September 25-27</td>
<td><strong>Agenda '88</strong> - Southwest US. The second annual... Run by Stewart Alsop, managed by Marketing Partners, sponsored by PCW Communications. Call Elizabeth Reademan, (415) 363-8080.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>October 11-14</td>
<td><strong>Info Show</strong> - New York City. Contact: Frank Fazio, Cahners Exposition Group, (203) 964-0000.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>October 23-28</td>
<td><strong>Monterey Classic</strong> - Monterey, CA. Contact: John Baumeister, (408) 987-4200.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Please let us know of any other events we should include. -- Anita Fowler

Release 1.0 25 November 1987
SUBSCRIPTION FORM

Please enter my subscription to **Release 1.0** at the rate of $395 per year in the U.S. and Canada. Overseas subscriptions are $475, airmail postage included. Payment must be enclosed. Multiple-copy rates on request. Satisfaction guaranteed or your money back.

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

Please fill in the information above and send with your check payable 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.

Sylvia Franklin
Associate Publisher

*Release 1.0*  
25 November 1987