ARCHITECTURAL CONSIDERATIONS: ALIBIS AND ADVANTAGES

In a world where 4 MIPS is passe and you can hang 700 megabytes off a pc, what's the point of mainframes? In a world where DOS can achieve "640K relief" and you can run workgroup applications on a UNIX pc/server, what's the point of OS/2? In fact, MIPS isn't the only issue. Nor is memory. Each of these architectures is optimized for a particular set of uses.

Contrary to the conventional wisdom, neither DEC's top-to-bottom VAX strategy nor IBM's Systems Application Architecture means that we will all be using the same line of machines, differing only in power and memory, for all our tasks. It simply doesn't make sense to do both transaction management and interface management in the same environment.

The key is not compatibility but connectivity. Ultimately, we are likely to have a number of standards -- or just a couple of standards per task. These will include architectures and configurations designed specifically for such tasks as data base management, transaction processing (a variation on data base management), inference or symbolic processing, text management, calculations. These functions will take place on servers, or what used to be called "hosts," but now they serve workstations rather than control crts.

Separate but designed to cooperate will be a rich, "friendly" personal workstation interface and a local application environment that allow the user access to all the server functions. (OS/2 and the Macintosh are the likely candidates.)

The goal of Systems Application Architecture is not so much to enable all applications to run on all machines, but to ensure that applications can talk to each other, sharing data and even tasks, calling on each other to perform the subtasks for which each system is best suited. For example, mainframes aren't
very good at managing user interfaces, but they are terrific at controlling multiple sessions and fast, frequent disk accesses. Even in the network/server architecture model, mainframes hung off a network can operate effectively as transaction processors controlling disk resources.

But why do we need mainframes at all? Why has Sun Microsystems, for example, just agreed to purchase a couple of IBM mainframe clones to run company administration and financial applications? Why couldn’t they just use a 10-MIPS workstation (equivalent to a VAX or a small mainframe in MIPS) or several 10-MIPS workstations lashed together with a distributed database?

That may be feasible someday, but for now the software just isn’t there. (Yes, there are stories of people replacing a mainframe with a large number of micros, but with exceptions such as Echlin Inc., which did a lot of in-house work to set things up, their mainframes were typically used for transaction-light tasks such as end-user modeling.) The distributed database required is still just a shimmer in the vapor, as are the applications, the application tools, the systems software utilities. Nor are robust transaction-processing environments such as CICS, TPF, or IDMS-DG widely available on anything other than mainframes (including Tandems and VAXen). The big task -- and forte -- of mainframes is disk management, something micros just haven’t been well configured to do so far. Mainframes’ I/O channels make the PS/2’s Micro Channel, a first step towards management of multiple I/O users, look very micro.

Whatever the underlying chip technology, there will continue to be a need for "mainframes," systems configured to handle large data bases with a high ratio of I/O to calculations. Their job is just to shuffle data around: It doesn’t take a lot of processing power, but it takes a lot of I/O bandwidth and good real-time resource allocation. Long run, there’s no reason you couldn’t optimize a 386- or 486- (or 68040-) based system for I/O-intensive transaction management, but you’d have a different beast from the PS/2.

**UNIX vs. OS/2**

The issue of Why mainframes? concerns time and semantics (although there can certainly be arguments about pricing, or about the capabilities of any particular mainframe or replacement therefor). The issue of UNIX vs. OS/2 certainly depends in part on timing -- if an 80386-specific OS/2 were available now or if a unified UNIX had been available a few years ago there would be little contest -- but it also reflects the capabilities of two specific products that implement the different approaches of two schools of thought.

Three broad factors favor OS/2: Its programming and end-user facilities are at least as good as those in UNIX; beyond that, it’s a matter of taste. It is better defined as a standard. And it’s the IBM standard.

(For the record, there’s no need to use either OS/2 or UNIX if all you want is to build or run the same old applications. Anything that can run under DOS will do much better without the overhead of OS/2 or UNIX.)

**Programming facilities**

Like a mainframe environment, UNIX is good at handling multiple users. By contrast, OS/2 handles a single interface and multiple processes, but even
operating as a server application it doesn’t need to know about individual users. In this way, it avoids a lot of overhead. (The needs of individual users, specifically screen and session management, are handled by each individual’s copy of the OS, whether it’s DOS or OS/2 or whatever.)

OS/2 differs from UNIX in its notion of a single user doing multiple things vs. UNIX’s many users who all have to be kept separate and out of conflict. OS/2’s architecture lets several applications or users have access to the same data (shared memory) and dynamic-link libraries (for sharing of code/applications) under proper control. Because multiple applications can share, you get data integrity, easy interprocess communications and software module consistency across applications, to say nothing of efficiency of memory use. By contrast, UNIX until recently rigorously avoided memory-sharing, reflecting its notion of users and program spaces that must be kept apart to avoid conflicts.

ACTIVE GRAPHICS FROM LOTUS

Is "interactive" so wonderful? Or is it just a polite way of saying user-intensive? Why should users have to tend their software all day? In the future, applications will be able to take care of themselves. For a little sneak preview of the kinds of things interprocess communications will make easier (familiar within large mainframe applications) and why we are all waiting for OS/2, take a look at Lotus’s new enhanced graphics package.

Aside from a variety of new charts and other handy features, and the combination of content and form analysis it provides, the exciting aspect of Lotus Graphwriter II is its ability to do its work automatically. If you update the spreadsheets, it can automatically combine them, follow a style sheet, and print out the reports you want. Lotus can do this without OS/2 because it owns the single application necessary to get the job done, and it uses standard file formats such as its own and dBASE. In the future, with common standards for communication rather than just file-sharing among applications, you may be able to have your WordPerfect text processor talk to your Lotus spreadsheet and your Microsoft dbms to gather the right data, produce appropriate letters, lay them out with Ventura Publisher stylesheets, and have them all waiting for you to sign when you walk in Monday morning. For that matter, maybe you needn’t show up until Tuesday.

In fact, OS/2 encourages carefully mediated "conflict" -- or interprocess communications. OS/2 offers several major interprocess facilities -- semaphores, signals, pipes, and queues. (All forms of UNIX have some of these and UNIX System V.3 has all of them, but implementation details vary.)

For those who care: Semaphores are simple flags that let one process know when another is finished, or when a resource is available, familiar to most programmers from the networking world. Signals are a limited form of message used to warn a process when something unexpected (such as a control break) has happened. Pipes allow the output of one process to flow directly through to be the input of another process; both processes are assumed to be running at the time the pipe is created and used.

Release 1.0

21 September 1987
In essence, a pipe is a direct, transitory link between two processes, while a queue provides for more general, controllable communications across time. It's like the difference between a phone call and a store-and-forward message system that can collect multiple messages for a single user. A single queue can contain input from several processes for use by a single receiving process (and of course the OS can manage multiple, named queues for any number of receiving processes). Unlike the data in a pipe, the data in a queue can be retrieved in specified order, and can be examined (peeked at) without being removed or deleted. (Facilities such as Dynamic Data Exchange depend on these kinds of interprocess communications, implemented in rudimentary fashion within Windows and slightly better in Windows 2 and 386. These facilities will be pulled back into their proper place in OS/2, leaving Presentation Manager to handle graphics issues only.)

Finally, there is graphics capability, a facility that gives OS/2 an edge in the single-user category, as opposed to the capabilities outlined above, which operate in the server/multi-user area. OS/2's Presentation Manager, whatever its intrinsic merits, will offer developers and users a consistent, well-defined interface that will insulate developers from differences across machines and peripherals, and users (so far as possible) from unnecessary differences across applications.

A single standard

What we see on single-user UNIX workstations is beautiful; unfortunately, the implementation varies from machine to machine and version to version. Part of the reason for the Mac's success is the highly defined strictures it imposed on software designers; OS/2 will offer some of the same benefits. OS/2 Extended will offer those, plus a dbms structure (a version of which will also appear on the Macintosh, courtesy of Sybase).

So what are the problems with UNIX? Unlike OS/2, it already runs in a 386-specific version. UNIX V.3, the emerging standard, offers a rich, powerful programming environment, complete with queues and most of the other features in OS/2 described above. If only the Xenix/UNIX V.3/Berkeley unification efforts had started a little earlier, things might have turned out differently. UNIX was built at a time when processors were expensive and end-user graphics were preposterous, and so it sacrifices human comforts to conserve what are now no longer scarce resources. As it is, UNIX proponents love the system, and point out that you can do almost anything in UNIX, certainly anything that OS/2 promises.

But that's precisely the point. Precisely because it does belong to somebody who will guard it closely in a way that AT&T did not guard UNIX, OS/2 will be well-defined (although broadly licensed). The great value of OS/2 is that it does things in a particular way -- which means not that different applications can operate simultaneously and independently, but that different applications written independently and not simultaneously can operate simultaneously and in tight cooperation. That is something you will never get with all the wonderful schemes to bypass memory restrictions, use 32-bit instructions and the like around and on top of DOS. That is something you won't get for some time from UNIX. Various studies show that only half of the functions in Berkeley 4.2 and UNIX V.3 operate the same. (Indeed, we had great trouble in preparing this article, because no two sources would agree on precisely what features UNIX does offer.) UNIX proponents point
out that UNIX is widely portable across machines. OS/2 proponents counter that "portable" is not "identical." And because OS/2 is Intel-architecture-specific, it will make better use of that architecture (ultimately) than a portable operating system ever could. Most people who buy OS/2 wouldn't think of moving to another machine anyway.

The IBM standard

Life isn't fair. One chief technical officer said to us bluntly, "I haven't studied the problem in detail because this company made up its mind early on to standardize on OS/2." (His marketing type recoiled in horror.) But Mr. X wasn't as careless as he sounds. His system is designed to work with other people's software, so he worries less about the problems he may have to overcome in dealing with OS/2 than about the problems he might encounter in connecting to other software, including DOS packages. Using a standard OS will be a good start. The fact that OS/2 comes from IBM/Microsoft, not IBM alone, gives users and developers the assurance it's not proprietary and that third parties will support it and rush to correct flaws it may have.

The reality

OS/2 should be available by mid-1988. Early reports of its robustness and speed are encouraging. True, next year is later than everyone hoped, and so is the timing of a 386-specific version, offering a large, flat memory address space, which won't show up until mid-1989. But it's early enough; we don't see developers moving wholesale to UNIX (which UNIX?) and OS/2 is a stronger standard than anything else. By its very newness, it provides an inviting market for new applications and new vendors. Standards aren't supposed to be perfect. They're supposed to be a reasonable compromise.

ROUND UP THE USUAL SUSPECTS

Imagine this company: Funding from Kleiner Perkins; chairman: KP's John Doerr, a key figure behind Sun Microsystems, Symantec, Compaq; president: Jerry Kaplan, a founder of Teknowledge and the man who developed Lotus's next hot product; vp of software: Robert Carr, founder of Forefront and developer of its and Ashton-Tate's Framework; investors and board members: Mitch Kapor and Vinod Khosla, founder of Sun and Daisy and also a partner in Kleiner Perkins; location (pro tem): Kaplan's house on the curvy part of Lombard Street in San Francisco.

It's no dream; it's GO Corporation, a mysterious start-up devoted to a "new personal productivity application." It has no formal relationships with Lotus or Ashton-Tate, but its founders have left those companies on friendly terms, and with promises not to compete. The company won't say more, but we have Kaplan's personal assurance it's not another spreadsheet. Indeed, GO aims to create/embodi a new category, much as the product Kaplan and Kapor have built for Lotus will do. As part of the growing Kleiner Perkins corporate combine, which also includes Sybase (page 25) and Metaphor Computer Systems, GO is a brand-name company that won't have much chance to make its mistakes in private. However, its founders have taken a vow of silence for the next couple of years, except for top-secret discussions to round up third-party support for the GOware. Will it use "artificial intelligence?" we asked. No, says Kaplan, "We'll use the real thing."

Release 1.0 21 September 1987
TEXT TOOLS: THE JOY OF LINKING

A recent flood of hypertext products -- personal hypertext tools\(^1\) -- has made hypertext a household word. Yet to come are industrial-strength hypertext tools, which will appear as the market shifts from one where only a few select people understand the issues to one where the customers want to buy the tools and do it themselves, thank you. This article addresses not hypertext itself but the tools that can be used to examine, classify, and structure text -- in a word, to find or define the kinds of links that hypertext can represent.

Direct manipulation. Freedom to associate. WYSIWIG. All of these notions reflect the power of a computer to do exactly what the user tells it to -- but no more. Direct manipulation requires that the user move things around on the screen. Hypertext requires that the user hand-build links between items. This is "intuitive," and it gives the user a feeling of great comfort, but it's a waste of the computer's talents, and ineffective for the building of industrial-strength, large-scale hypertext systems.

In short, hypertext is the representation of the results of intellectual activity; it is not a tool that automates that activity. It allows a user to follow unstructured links within a base of text or other data (HyperCard's "cards," or data records, for example), but it does not control the creation of those links. The user must determine where those links belong.

Of course, you could automate the process by, for example, instructing the system to create a link among all uses of any particular word -- but you'd get a meaningless morass. Like a sculptor, the hypertext builder applies his intellect as much in peeling away extraneous links as in implementing "good" ones.

Hyper(con)text

Hypertext is like any buzzword: It's a vague reference to a concept -- in this case, links between data -- that can be implemented in many different ways and as part of many different applications. Here are just a few approaches to hypertext:

- **Hypertext as a personal tool, for the user to develop his own world.** HyperCard, Owl International's Guide, and Knowledge Systems' KMS are all fine examples. The user can create text (or other data) nodes, and link them to each other. By selecting a marker (typically a "button" or a highlighted word) on the screen, the user can follow these links from node to node.

- **Hypertext as a finished product, a pre-linked selection of texts that the user can explore by following the links developed by the hypertext-builder and (usually) by using text-search functions.** Hypertext is the embodiment of data, plus the intellectual effort of people who see connections among small or large chunks of the data. Grolier's Electronic Encyclopedia and Symbolics' on-line documentation are examples of hypertext as medium.

---

\(^1\)For an excellent survey of projects, products, and issues (but not the tools covered here), see Jeff Conklin's excellent survey in the September issue of IEEE Computer; he's listed in Phone Numbers, page 26.
"Executable" hypertext, or hypertext with meaningful links that can be reasoned about by a program. This rich form of hypertext has "typed" links, so that it can be reasoned about, as opposed to being an inert file with links in it. In some ways richly structured hypertext is akin to an object-oriented data base or even to a program, depending on the capabilities of the links.

Hypertext tools, text analysis and structuring tools that assist in the development of hypertext (and that can also be used in interpretive mode for exploration by end-users). Obviously, there's a range of not very rigid capabilities and positioning here. Up to now, the generation of hypertext has been an activity engaged in mostly by custom houses such as KnowledgeSet, Knowledge Access and Quantum Access, and by electronic data base providers such as SilverPlatter and BRS Information Technologies. They do custom preparation of text for traditional modes of on-line access, including structured fields (e.g. periodical data bases with fields for author, title, subject, date, etc.). These vendors have traditionally been considered part of the CD-ROM or online business, rather than the text-structuring business.

And the hypertext workbench, a set of such tools positioned explicitly for the creation of hypertext. Quantum Access sells QA Gateway, and KnowledgeSet will be entering that market shortly with Desktop Data Prep.

The hypertext workbench is not a system to "create" hypertext, but one that will help a user to do so, in much the same way that a programmer's workbench doesn't actually produce code, but helps the user who is doing so. The workbench lets the hypertext builder look at his data in useful ways, with automatic indexing of words and phrases, synonym substitution, and subject matter hierarchies. It lets him inspect a chunk of text and derive its topic, or its similarity in topic to other chunks of text. It lets him parse the structure of text -- headlines, indices, tables of contents, bibliographies, "see also"s -- both to find links and to determine content and create structured fields. Using these tools, the hypertext workbench can automatically classify and define text, and generate links from which the hypertext builder can instantiate the ones he considers valid or relevant.

The index is the foundation

What sorts of things do the hypertools do? First, they index the text every which way, creating a set of inverted files. An inverted file is simply an alphabetical listing of every word in the file except (optionally) for designated "stop" words such as "a," "an", "the," with a pointer to where each can be found in the text (that's why indexes take so much space). Indexes can list words, phrases (using special phrase-finding algorithms), or key words in context (KWIC), among other things. A KWIC index lists phrases by key words -- that is, it ignores word order, as in "codes, building; codes, moral; codes, secret; etc."

These indexes are used for simple word searches, and for Boolean queries. Say you want to find "Juan" within 25 words or fewer of "Alice." The software will look in the index to find the locations of all references to Alice and compare them to the references to Juan, and return the pairs with
locations that match by 25 words or fewer. (All this can be done with the index alone; the text need not be examined.)

Then there are hierarchical indexes, of which the most familiar are the table of contents and the outline. Like alphabetical indexes, these have a particular structure that enables users to navigate them easily, and then follow a pointer from the selected word or topic to the text that contains or refers to it.

CD-ROM and hypertext

CD-ROM and hypertext fit together like peanut butter and jelly. They are not the same thing, nor are they technologically interdependent. But they are frequently found together, and they enhance each other. Both are media for information: One, CD-ROM, is hardware that holds the information; the other, hypertext, is software that defines how it is structured. You can have hypertext on magnetic media (or even on cardboard notecards tied together with string, if you insist). And the data on CD-ROM can be structured in as many ways as the data on a hard disk, including flat files, object-oriented data bases, relational tables as well as combinations of these linked together with pointers -- that is, hypertext. Because the amount of data stored on CD-ROM is so large, because much of that data tends to be text, because hypertext is one way of making large amounts of unstructured data accessible to end-users, and because both these technologies are rapidly approaching commercial viability at precisely the time that end-user accessibility is a hot issue...for all these reasons, hypertext and CD-ROM fit together like peanut butter and jelly.

Technologically, CD-ROM is fixed, read-only. But while most CD-ROMs carry their own inverted file indexes, hypertext data access structures -- thesauruses, queries, meaning hierarchies -- can be kept on magnetic media and enhanced as appropriate to provide flexible, updatable access to the fixed data on the CD-ROM. New associations not implemented in the fixed CD-ROM data can be incorporated into the hypertext front end. Moreover, hypertext is uniquely flexible in its ability to address a variety of media (hypermedia). For example, you could change the front-end of a CD-ROM data base so that selections on certain links are diverted to updated information in another medium, to a sequence on a VCR, or to an application, as in HyperCard. (A comprehensive survey, Optical Media Systems: Textual Retrieval and Networking Issues, is forthcoming shortly from John Gale’s Information Workstation Group.)

There are two further embellishments. One is a form of hypertext itself, an index structured by content rather than by the alphabet. A thesaurus index is such a system, with related concepts linked together. Once the user finds the concept he wants, he can then move from it across links to the related text. A more hierarchical topic index takes the form of a tree or taxonomy of related concepts, such as an organization chart or the tree of the animal and plant kingdoms that we all learned as children. Such an index is much easier to browse than the much larger base of associated text to which it contains pointers that you can use once you have selected a topic, and so it provides a useful front-end. An example is AIRS’ Marcon.

Release 1.0	 21 September 1987
A second enrichment of index facilities is to have each topic in the index represent not a pointer (or set of pointers), but instead a query that can be run against any text base. Instead of having fixed hypertext links from index to text, you can create new ones automatically. In other words, you can update the text base and bring in new information without having to alter the index; the index can find the appropriate new information automatically. (In practice, the query is typically run against inverted files rather than the text files themselves, but the creation of the inverted files is a simple, mechanical process, if sometimes slow.) An example is ADS's Topic, which combines a hierarchical topic index with automatically generated queries. (A similar facility could also be built into a hypertext text rather than its index -- a fine distinction. That is, at point \( x \) in the text there could be a query that would select text that meets certain criteria. The most current text would then be linked in at runtime.)

These facilities can be used both to build hypertext and to explore it. They are not what everyone thinks of as hypertext, which is considered to consist of pre-formed links between text nodes, but they are necessary for the creation and exploration of large text bases. In practice, it will be a long time before most text is fully cross-referenced, and these indexing tools give users as well as builders the ability to classify text and navigate it easily.

**Defining topics and queries**

Once the words are indexed, they can also be used to define the text that contains them. In the simplest example, an article containing the word "cats" several time is likely to be about cats. It could also, of course, be about mammals, about pets, or perhaps about tigers or theatre or jazz musicians. How do we know? (Or how can the software know?)

The presence of other words, such as "jungle" or "carnivore," or "stage" or "saxophone," could provide clues. The value of the "leaves" in the topic tree in classifying a particular item can be changed by assigning each a weight: for example, the presence of the word "credenza" is fairly strong evidence for an item about furniture, whereas "table" and "chairs" offer less sure evidence. (It could be an article about meeting procedures.) You can also give words negative weights; in this case, "leader" or "meeting" might weaken the evidence for furniture, and strengthen it for meeting procedures. Then, by hand, with the aid of a thesaurus/synonym finder, a builder can add to the hierarchy, adding "felines" to "cats," for example.

Using these methods, the builder can create a large number of keyword-based combinations of topics and queries, each of which can be run either to find a piece of text or to characterize it as it is indexed on entering the system (imported from another data base, or typed in and stored, generally).

Figuring out appropriate synonyms and co-occurring words for one word is fun; figuring out an entire thesaurus, or an entire hierarchy/taxonomy of words that subsume or encompass certain concepts, is a monumental task. Fortunately, rudimentary forms of some such structures already exist.

For standard texts and text bases, there are thesauruses available, such as the one for the National Library of Medicine's Medline data base. Many standard reference works have rich multi-level indexes that can provide a
basic hierarchy of concepts that can be applied not only to the work in question but to other works of similar nature. That is, an index to one biography of John Sculley would provide all the concepts covered in another biography of Sculley and most of those in one of Steve Jobs, although the precise locations (electronic or page) would of course vary. Other hierarchies could be drawn from a biology textbook (for animals) or an outline or table of contents (which are the same thing in different form). (See also our discussion of the Persoft product, since named IZE, in Release 1.0, 12 May.) In the worst case, such a hierarchy must be built by hand, or by mind, by a subject-matter expert.

Plain text, structured text, and hypertext

Obviously, the richer the structure of the incoming text, the easier it is to parse it for hypertext. Chapter headings and subheadings, references to figures, diagrams, tables and other sections of the text, indexes, footnotes, all help to provide preinstantiated links for the hypertext developer. There are other less universal tags, too. For example, Grolier's Encyclopedia capitalizes all words that have their own separate entries, which makes it easy for a developer such as KnowledgeSet (which did the job) to link all these references automatically to the appropriate articles elsewhere in the encyclopedia. (Just watch out for double meanings like AI and AI, however!)

While there are few standards as yet in the hypertext world -- we're at the point where products are proliferating, not converging -- Standard General Markup Language, an emerging standard, provides a cross-environment means for text-preparers to create and represent links. SGML is well-known as a way to indicate type sizes, positioning, and formatting for electronic publishing output; less well-known are its facilities for indexing, cross-referencing, and anchoring pieces of text or other objects to one another. These links can then easily be read by most text-preparation systems.

Defining text

How else can content be defined? Frequently, you may not have a perfect, rigorous description in mind, but you may have an example of what you want. For instance, that's why we found a lot of background material for this article under "KnowledgeSet," some more under "text dbms" and "text search" and "CD-ROM," and yet more under "Brattle." Suppose we could have fed a couple of articles from each of those files into a system, and told it to go look for others like them? That's a capability offered by Personal Library Systems, using a combination of statistical and heuristical techniques that the company won't discuss in detail.

Other ways of defining text are very clever, if low-tech. The distinctions between headings and plain text are obvious. An article with the words "French history" may not be about French history, but one with those words in the title probably is. You can continue by assuming that articles by the same author cover the same topics. Likewise, articles cited by another article probably cover related topics; you can determine the topic of an article by examining its citations. All of these are "tricks" that work on common-sense assumptions -- the same sorts of assumptions that a smart, knowledgeable librarian might make. Of course, the software can't infallibly determine the topic of an article without understanding it -- but neither could we. (Taken any reading comprehension tests lately?)
Some problems are pretty low-tech, too. For example, a certain KnowledgeSet client requires that no data be changed, just in case what looks like a mistake is not one. So, the text base contains about 6000 uses of the word "hydraulic" -- 5669 spelled that way, and a few spelled each of six other ways. In dealing with legal text bases, many experts throw out all words that appear only once on the assumption that they're misspellings.

In fact, it can be extremely difficult to build hypertext from existing text bases, says president Marty Kahn of BRS Information Technologies, an information vendor. First, the structure of the text itself is wrong: Hypertext works best in small chunks, no more than a couple of screenfuls. There's also a need for tremendous consistency (at least until we develop better fuzzy search techniques). You must work on the assumption that the material will continually be updated; authors of regular text tend to deliver their work and leave it at that. Recently, the people at BRS tried to turn some medical textbooks into hypertext and concluded that at least in this case retrofit hypertext simply doesn't work. They have moved on to building the text from scratch in a massive project involving 53 doctors at Columbia.

Here a link, there a link, everywhere a link link

Rich hypertext includes the notion of webs, or an individual's idiosyncratic view of a particular set of information nodes (much like a relational view of multiple tables). A psychologist will see one set of connections in a child's fairy tale; a social historian will see another; and the child will see yet a third. A brilliant historian or detective will likely see more and more interesting links than a lesser thinker. All these "webs" may be of interest, but some will be of greater merit than others. (Intellect and art are not egalitarian.)

Hypertext can also allow for the construction of different kinds of links -- supporting or contradictory evidence; to more general information, or to more specific information; to biographies, contemporaneous events, geographical background, maps, etc. These help to reduce the complexity and apparent formlessness of the links and the resulting web of links. Links can also be active or executable, defining relationships among nodes and resolving conflicts or voting on decisions. Yet another hypertext construct on the way is metastructures -- fileboxes, hierarchies, and other ways of aggregating nodes into collections that can be handled as a single object or broken into their constituent parts.

All these hypertext tools do is help create tags or labels for the text so that the user-builder can instantiate the proper links. Once the tools have done their work, the user's sensibility comes into play. In the end, building hypertext is very much a value-adding process, with the value added being that of a discerning intelligence that can make valid judgments about where and what kinds of links should be created.
GENERIC LINKS: SOME COMPANIES HYPERTEXT MAKES US THINK OF

ADVANCED DECISION SYSTEMS/VERITY SYSTEMS

Advanced Decision Systems is an AI research and development company that mostly does military-oriented expert systems work. One of its projects, with a working name of Topic, began as an expert system for analyzing text and evolved enough to acquire its own division, Verity Systems. From a LISP prototype on a Symbolics 3600, Topic now has grown its own language for manipulating tree structures, specifically topic hierarchies to define text.

To the hypertext builder, Topic offers a quick way of classifying pieces of text, based on a rich structure assessing evidence based on the presence or absence of certain words within the text. It can be used interactively as a search tool by an end-user searching for information on a particular topic within a text base, or it can be used in a production mode to classify all the pieces of text within a text base as it is built or as they are entered. Note that Topic deals with chunks of text as a whole, rather than finding a particular place in a long document. When the document is displayed, it will highlight the words used as evidence, but the actual words in the topic description may never appear in the text.

In essence, it uses the hierarchical structures mentioned on page 9. A builder/user who understands the subject domain builds the hierarchy, using whatever pre-canned hierarchies he can find, and assigns weights to the presence of certain words or phrases (the number of times a word appears in a given text is ignored, however). This hierarchy itself looks like a combination hypertext/hierarchy web. A single word can occur in several branches of a tree, and the absence of a word (NOT-X) can also confirm the likelihood of a topic match.

This tree structure now looks like a map of possible topics. In fact, it is also an executable program, and the nodes within it with their branches constitute smaller programs. A user can now select a topic, such as "cats" or "friends of Ivan Boesky." Topic then runs a program that consists of the branch of the tree extending from the node selected (although the user typically sees his options as an alphabetical list). The program runs against the inverted file for each text selection in sequence, matching the words in the file against the leaves of the branch. The matches filter up through the weighted structure, and various algorithms construct a value for the likelihood that each particular selection concerns the topic desired. The user can then view as many of the top choices as he wishes, or, if he's building hypertext, insert links to them (or refine the query further).

AIRS

AIRS, for Automated Information Reference Systems, was founded in 1983 by Ted and Fred Durr, former academics. With $1 million in private funding, Marcon does custom data preparation work for information sellers and also

21This could of course be expressed in traditional expert systems terms, with a rule and confidence factor for the presence of each term -- and it was! But that structure is fairly cumbersome. Topic's language, designed specifically for this task, works much better and more easily.

Release 1.0 21 September 1987
sells a text-search product, Marcon, which costs from $895 to $1495 in varying configurations on a PC. The 12-person company's major revenues come from this project work and from runtime sales of Marcon used to access the data in the systems it builds. These include a hypertext version of the Bible using 11,000 Strong's numbers (a standard set of word studies and cross-references); a 300MB water resources data base for the US Geological Survey, available in November; a robotics engineering data base for Loyola University; and a data base of environmental rules and regulations for Environmental Resource Management, which has resold it to 70 customers since April. (In a survey of 49 of them, only two ever looked at the manual, notes Ted Durr proudly.)

Also coming soon is AIRS' 440MB hypertext version of the National Library of Medicine's multi-gigabyte data base, Medline. The data in Medline is generally available to anyone who proves a certain level of credibility, and is currently licensed to about 14 organizations, of which only a couple have products on the market. AIRS considers its edge to be an elegant use of Marcon, which enables it to include a hierarchical hypertext index based on the Medline thesaurus. This front-end makes the structure and the availability of data easily apparent to the user, while the other vendors so far are simply offering full-text and structured-field search.

Marcon's standard text-search engine is enriched with a hierarchical meaning structure similar to Verity's Topic. However, the system does not itself derive the topics of the text (in the Medline data base, for example, they have been assigned by human researchers at the National Library of Medicine). That is, once you find the topic in the hierarchy, you follow a direct, hypertext link rather than a computed one to the appropriate piece of text. (As with Verity, the hierarchy itself is also pretty much constructed by hand.) Marcon's hierarchy can easily be permuted, so that instead of looking at children's diseases, including colds, you could look at colds, including both children's and adults' colds from separate branches of the hierarchy. You can also explode a query, getting references not just to a topic itself but also to a variety of subtopics. Of course, you can also use standard text search procedures for topics not in the topic hierarchy.

BRATTLE RESEARCH

Brattle Research, founded in 1982 by natural-language expert John Clippinger, offers a wide range of tools that can be used by anyone structuring or inspecting text, handy for building hypertext or other purposes. Brattle is working at the far reaches of text analysis, attempting to get closer to "understanding" the meaning of text than any of the other vendors described here. Specifically, the company is working on a project using The Wall Street Journal and other business publications that involves parsing sentences in news items to catalogue their contents in a structured way, determining the relationships among data items. For example, it can search sentences for individuals' names and match them to titles and companies by syntax, or it can compile reports based on news releases about company earnings or corporate take-overs. Reliability is still low enough (about 90 percent for simple cases) that the information generated may need to be checked by a human reader, but finding errors is a lot easier than compiling the data by hand (and is necessary with human indexers too). Separately, Brattle is working with Arthur Andersen to develop Who-What-Ware, for auto-
matic filtering and compiling of business information, and on another project with the New York Stock Exchange for surveillance of insider trading. In essence, while the other systems determine what a text item is about, Brattle comes close to determining what it says.

KNOWLEDGESET

KnowledgeSet was founded three years ago as Activventure by Digital Research founder Gary Kildall, who shares time between the adjoining buildings that house both companies in Monterey, CA. George Banta Co., a large U.S. printing company in Menasha, WI, that also has video production interests, has just bought a majority interest for an undisclosed amount. For the moment at least, the effect of the arrangement will be to give KnowledgeSet extra funding to carry out the kinds of activities it's already doing.

At this point KnowledgeSet is still mostly a consulting/custom work company, developing graphics-rich hypertext bases for large customers. Its most visible project was preparing the first version of Grolier’s Electronic Encyclopedia on CD-ROM. Its current projects are considered competitively strategic by its customers; they won't allow their names to be used, but they include a 250MB CD-ROM system (replacing microfilm) for aircraft technical documentation, including text and vector drawings. Vp engineering Tom Rolandar says the company hopes to do more royalty-based work along the lines of the Grolier project, for which it still receives royalties from Grolier's sales, and also to sell tools to customers in order to gain more leverage.

KnowledgeSet's tools will be available early next year in a pc-based package called Desktop Data Prep, to be announced this week at Link Resources' CD-ROM Expo in New York City. DDP (no price yet) will include a search engine for indexing and finding terms, a variety of indexing methods, a user interface toolkit, raster and vector image-processing tools, and a standard input filter for SGML tags. Industry consultant John Gale calls the company's software the "first second-generation hypertext software" he knows, with an intuitive interface unequaled elsewhere.

KnowledgeSet has noticed increasing interest among its customers in bringing some of the work in-house, as represented by strong sales of hardware CD-ROM preparation tools such as Meridian's. However, we suspect that KnowledgeSet will still be doing a lot of customer support and custom work for years to come. The qualifications are vague, but take it on faith: Understanding and manipulating text and graphics, which are so much richer than numbers and data, requires smarts and experience as well as tools and theory. (It's the same as what makes an expert better than an expert system.) Each project requires a great deal of custom work, ranging from understanding a client's way of structuring the data to defining domain-specific index terms. But each project also benefits from a company's breadth of experience with previous projects.

KNOWLEDGE SYSTEMS

Knowledge Systems' Knowledge Management Systems is an outgrowth of a Carnegie-Mellon project called ZOG. Released commercially last June, KMS sells for $845 per copy, relatively cheap for a Sun workstation product. Founder
Rob Akscyn is looking for a big market (by Sun standards), working on the assumption that he should get a substantial proportion of the users of Sun workstations, who tend to be both builders and users of software. Indeed, KMS is like a fourth-generation language in that it is both a development and an execution environment.

Knowledge Systems consists of president Akscyn, chief technical officer Don McCracken, and Akscyn’s 26-year-old brother Richard. Number-four employee Elise Yoder has just left on friendly terms to start Knowledge Workshop, which is using KMS for publishing applications. KMS offers rich output design facilities and direct manipulation of text and graphical elements -- i.e. most of the capabilities of such a tool as PageMaker.

Knowledge Systems has also put in facilities for multiple users, so that tasks such as e-mail work easily within the KMS environment. KMS’s e-mail is not a separate application but simply the ability to move mail into another person’s mailbox. If there’s a lot of mail, you can move it to another frame (node) with a pointer to it -- and you can restrict access to the underlying message if you wish. The recipient can then sort his mail and link it to appropriate topics. The metaphor is simple, the execution is fast, and the implementation is extremely rich.

The developers of KMS have flouted conventional wisdom in several respects, optimizing their system for performance and limiting such fashionable tools as overlapping windows, extensive menus, and the like. (They have published several lucid papers examining the trade-offs.) There are two kinds of links, hierarchical and cross-references. The system’s overall hierarchical structure, plus a fast text-search capability, makes up for its lack of a graphical browser or the kind of file-management system found in HyperCard.

KMS addresses a major complaint against hypertext -- lack of structure and ease of getting lost -- with its implicit hierarchical structure. Links go one way -- from a text element within a frame to another frame -- and back. That is, you can go down in a hierarchy (and you can create an interlocking set of hierarchies), but there is an implicit notion of each set of frames constituting a tree. (Any given frame can be part of several hierarchies, but if you go back, you go up into the hierarchy from which you reached the frame.) You can also use the Goto command and move to a frame you specify by its ID number, its hierarchy name plus the number of the individual frame. Or you can search for any particular string, starting down from any place within a hierarchy to limit the scope and speed the search to the extent that you can.

The hierarchical structure, plus an avoidance of embedded links (which amount to long parenthetical phrases) constitute an advantage in the generation of paper print-outs, but are less well-suited to some other kinds of applications. KMS lets users see two nodes at a time, which is less than twice as good as one node at a time, but certainly almost as good as three, four or more at a time. Within the Sun operating environment, moreover, you can also run other applications within other windows.

Interestingly, KMS/ZOG began as an exercise in building a “human-computer interface system.” But the developers are open-minded enough to acknowledge that “experience with ZOG and KMS has convinced us that the data model underlying an interactive system is more important than its user interface.”

Release 1.0  21 September 1987
Perhaps the ultimate purpose of a good interface is simply transparency and focus -- making the system's capabilities clear to the user and presenting the most appropriate functions at the most appropriate time. KMS's only "curlies" are its three-button mouse and a customizable menu for system functions at the bottom of each window.

The creators of KMS note that the best way to understand the system is to move within it and help create it; their notion is very much one of active user-builders. This is an inspiring notion and it indicates tremendous respect for users, but unfortunately it's not appropriate for all users.

MCC

Some of the more interesting work on hypertext is going on at the Microelectronics and Computer Technology Corp. (MCC), where NoteCards co-designer Frank Halasz is developing his ideas further. From his new vantage at MCC, the former PARC researcher is taking a dispassionate view of NoteCards' (and other hypertext systems') deficiencies and the challenges/opportunities they offer. Specifically, Halasz and his colleagues Jeff Conklin and Michael Begeman (the "hyperactivity team") are addressing issues of search and query, structure (hierarchies and collections of nodes), version management (what happens when a linked node changes or disappears?), typed links, support for collaborative work, and extensibility. One application for many of these features is "issue-based information systems," to aid in making design decisions for large software systems. It could also be applied to other ill-formed problems, such as urban planning or whether to build a new airport in Austin, says Halasz.

MCC, Xerox PARC, Tektronix and Brown University are all hotbeds of hypertext research and experimentation, although Brown's concern is more with the incorporation of other media (hypermedia) than with the structure of text alone. Much of this work will be discussed at a workshop this November (see calendar), which will result in a book available next year.

PERSONAL LIBRARY SYSTEMS

Another approach, used by Personal Library Software, an offshoot of the Sire (Syracuse Information Retrieval Experiment) project at Syracuse University, offers one of the industry's more exciting topic-definition and query-generation capabilities around. In essence, PLS's Personal Librarian offers "symbolic query by example." You give it an example of a word or a text you like, and it proposes other words or texts you might like. The process works thus: A user suggest a keyword, and the system inspects the text base (its inverted file, actually) to discover other words that occur near it throughout the text base with greater than usual frequency. These are then proposed as "expanded keywords" that the user can use as the basis of a search, selecting the ones he considers relevant.

Alternatively, the user can select one or more articles to use, and have the system use all the words in them for querying the rest of the text base. Or, if he doesn't mind waiting a few minutes, he can expand all the words in the selected article, and use all the expanded keywords as the basis of a query. The system delivers the articles selected ranked by the closeness of
the match, or "probable relevance," depending on criteria such as frequency of the words in the selected texts and their frequency in the text base overall. At any point, he can select a text to read, or accept some and reject others and make the system further refine its definition.

PLS and a predecessor company have sold hundreds of copies of Sire/Personal Librarian since 1984, says founder Matt Koll. Written in C, the product runs on pcs ($600), as well as multi-user UNIX systems and other machines ($5000 and up).

QUANTUM ACCESS

Quantum Access has just started shipping QA Gateway, a $10,000 pc software toolkit for CD-ROM indexing and retrieval, with a development library and interface builder for $5000 more. This price includes up to a week of training for three people plus unlimited phone calls. "For a fairly standard data base, we can show them how to use it in a few hours, but if you want to get fancy there's a lot of art in there. Embedded in the software we have an awareness of the hierarchical structure of data, with a dynamic, ThinkTank-style outline," says ceo Draper Kauffman.

Quantum Access, primarily a service bureau/software provider, has used QA Gateway in-house for some fairly fancy work, including a 300MB data base of regulations for the U.S. Air Force, rich with cross-references and graphics, that will be released next month. In that project, says Kauffman, the company combined the tables of contents of all the constituent texts into one gigantic hierarchy which it subsequently restructured four ways. The text base is now accessible four different ways -- by topic, by publication category (DOD regulations, Air Force specifications, etc.), alphabetically by long title, and in a tree-structured user guide organized by frequent user questions.

Formerly technology planning manager for the Houston Independent School District, Kauffman entered the business with three partners by building a 120MB text base of school regulations and administrative documents in 1986.

XEROX PARC

Xerox PARC is one of the few outfits working seriously on the notion of typed links (the other is MCC, in a project run by ex-PARCer Frank Halasz). Xerox developed one of the first hypertext systems, NoteCards (Release 1.0, 31 December 1984), which finally inched to commercial availability as an unsupported (and basically unmarketed) product last year. Commercial customers are using NoteCards for such tasks as maintaining information about customers and competitors; interest from prospective customers has increased sharply since the announcement of Apple's HyperCard. Internally, it has been used to build Xerox's Instructional Design Environment, which helps a user from the point of setting up instructional goals through designing a video-based training the course and developing a prototype for ultimate delivery on a pc.

Designed for Xerox's LISP D-machines, NoteCards is a rich system that allows a viewer to see an arbitrary number of cards on the screen. It has some

*Release 1.0* 21 September 1987
notions of structure, including Fileboxes (or collections of cards), but its text-search capabilities are limited. NoteCards also supports the creation of different types of links. Take the links between components of a task, for example, or the different links among the A, B, C, D nodes in a web of information where if A and B are true or if C is true, then D is true -- unless E is true. Or perhaps A supports the notion that D is true, but it's not incontrovertible evidence (that's much more realistic, isn't it?). Other relationships include links to source material (or citations), further information, such as maps, biographies or other cross references, reasons (why or how something is true as opposed to whether), questions, and so forth. Such a rich hypertext system can form an executable knowledge base, as data filled into one node ripples through a web of structured links (although the software to execute it is not in NoteCards itself).

The structured links, which de facto make the nodes into typed data with associated procedures, can then be acted upon by an application. For example: Collect all the steps required to fix any part of the machine that distributes toner (the sort of application that might come in handy around Xerox). Or, it turns out that Henry was at a product-launch party the night of the murder and that Juan did not know Alice; what hypotheses for the true identity of the murderer do the new facts support?

IN MEMORIAM

TOM LAWTON
NITTY-GRITTY EXPERTS: AMERICAN EXPRESS RELEASE 2.0

Much of the glamor surrounding expert systems has dissipated (see next page). Skeptics have heard certain examples of expert systems' productivity so many times (DEC's XCON and the Campbell soup cooker come to mind) that they've come to doubt the existence of any other examples of practical, implemented expert systems. In fact, there are many, but their users tend to keep them quiet because they constitute a competitive advantage. Thus we're delighted to add one more tangible example to the line-up...

Just a year ago we wrote about American Express's Authorizer's Assistant project, which uses Inference's ART expert system tool to enhance the performance and productivity of the 200-odd people who authorize specific card purchases (as opposed to the credit evaluators who decide whether to issue cards to individuals). The project will soon dissolve -- into an actual production system, based on some heartening number-free but ratio-rich findings American Express is releasing. They show that the use of expert systems can reduce credit losses as well as increase internal efficiency.

"From an efficiency perspective," the American Express report says, "the AA will autonomously handle between 20 and 35 percent of the current [authorization] traffic, and contribute to a 20 percent reduction in the handling time for transactions that go to authorizers. These productivity data alone produce an internal rate of return between 45 and 67 percent."

Of course, notes American Express vice president of technology strategy and project instigator Bob Flast, "we've known about the efficiency impact for months." Far more interesting is the revenue, or loss-control, impact. Based on the data so far, says Flast, "for the approved transactions, where a small fraction today wind up in trouble 90 days later, AA advice should result in fewer cases in collections by 50 percent. For the declined transactions, AA advice should identify 100 percent more cases destined to wind up in collections than the current manual procedures. The losses avoided by this improved screening are more than five times the productivity savings." In addition, about a third fewer transactions can be declined, while still producing that reduced loss rate. "That's the beauty of this thing," says Flast. "Service goes up while losses go down."

These figures are stunning. But their mere existence also reflects another implication of the adoption of expert systems. "For years," says Flast, "companies have been hiring and promoting decision-makers without ever objectively measuring how good they are at what they're supposed to do. We've measured how well they follow the rules, but we haven't measured the rules themselves." At first, American Express simply gauged the performance of the automated AA against the performance of the company's senior -- and ostensibly "best" -- human authorizers. But to measure the AA's performance, the company started taking a look at subsequent-loss factors, and has discovered the comforting news that the rules are in fact valid; the more closely the rules are followed, the better the service and the lower the losses. (American Express is a well-managed company; we suspect that in other cases the introduction of expert systems will encourage measurements with results that may motivate management to reexamine the rules their employees are following.)
One source of resistance to the project, notes Flast, was the formerly well-founded fear of computer systems' rigidity: "Whatever we build better be good, because we're stuck with it." In fact, expert systems are far more flexible than traditional dp systems. Moreover, we'd like to point out, expert systems may even be more flexible than people (or neural nets), because they don't form habits; as soon as you change the rules, they'll adopt the new ones without a murmur.

Based on this experience, American Express is working on several new applications of artificial intelligence, which it will discuss in due course.

NITTY-GRITTY EXPERTS: PALLADIAN RELEASE 2.0

Palladian has recently undergone a management change, reflecting some disappointment at the company's sales rates of its Management (formerly Financial) Advisor and its Operations Advisor. Phil Cooper has resigned as president and ceo positions, although he remains on the board. Former marketing and sales evp Jim McGowan (from IBM before that) has become ceo. The company's products are rich, powerful systems that run in a LISP environment (although MA has recently been ported to Apollo). Both faced a tough market at prices closer to $100,000 than to $10,000, and the company dropped the first-unit price of Management Advisor from $66,000 to $15,000 two months ago. Operations Advisor is still at $90,000. MA, in particular, faces lots of competition from less expensive systems: How much more valuable is a "yes" backed up by LISP reasoning and elegant graphics than a "yes" backed up by a PC-based spreadsheet model and a couple of charts? In a world where the impact of financial and planning decisions is time-lagged and hard to measure, MA's worth is not immediately apparent. By contrast, the output of Operations Advisor, which simulates the flow of work through a factory, is far harder to emulate on a smaller system and provides tangible, specific feedback on factory or paper-shuffling operations. (See Release 1.0, 31 October 1986.)

McGowan does not plan a wholesale refocusing at the company, but he notes that staffing is down to 50 people from 92 when he arrived last February. Perhaps the biggest shift will be increased focus on connectivity to outside data sources (spreadsheets, data bases, operational figures), in response to customer feedback, and lessened focus on AI. "The first wave of people buying AI for its own sake is over," says McGowan. "Expert systems is the way to build it, but functionality is the way to sell it."
THE VALUE IS THE DATA

Long ago we wrote an article (31 December 1984) limning the migration of the value-added from hardware to systems software to tools to applications to data and market- or function-specific software. In short, as each sector blossoms, the offerings ultimately turn into commodities, and the value-added slips further out into an area less well-supplied. This movement is forced both by technological progress and by marketing imperatives, as vendors search for a way to enhance and differentiate their products. Although everyone wants to create the standard, the truth is that the economic returns from those standards typically diminish once the standard is established. (Just consider the history of IBM’s PC.) This is not an even process, of course, but it’s clearly visible, and it has something to do with why Apple is offering HyperCard and IBM is offering OS/2 Extended as part of their operating systems suites.

Further down the spectrum, much of the PC tools business is now losing its edge. Just how much value can you add to a spreadsheet (although a lot of people will be trying this fall!)? Likewise, there’s a proliferation of file managers, games, etc. Most of the new crop of agenda managers (with one notable exception to be discussed later) are simply restricted versions of data bases, with some files preconfigured and some reports pregenerated. The vendors have done much of the work for you, but frequently they have also restricted your access to the full power of the underlying data base management capabilities.

I-TRACK: MARKETING IMPERATIVE

What made the Barbie Doll so successful? No doubt it was her wide range of outfits, everything from nurse and stewardess to (more recently) travel agent and country singer. Little girls who lacked imagination could rely on Mattel’s ideas, and little girls who had imagination could build on them.

That message hasn’t been lost on I-Track Corp. founder Bill Spear, who says he keeps a copy of John Naisbitt’s Megatrends under his pillow (for better or worse!). His product Inside Track II could be this fall’s Barbie Doll. A simple, inflexible file manager, the $100 product offers value-added by way of a growing series of $40-and-up floppy-based InfoGuides, including names and phone numbers for any of a growing list of 40 cities, industries, etc. Not only does this provide value to users, but it also gives them an indication of how to use the product. Coming up (at users’ requests) are data bases for golf buffs, rock fans and other enthusiasts. An InfoGuide version of Egil Juliussen’s paper “Guide to the Computer Industry” sells for $49.95. Imagine the appeal of Donna Rice’s little black book, Ollie North’s fund-raising contacts, the Comdex Guide to Las Vegas, or our very own guide to hotels with pools (listing hours, length and shape, temperature). You don’t need CD-ROM to use software as a publishing medium.

The product itself is likely to whet users’ appetites for something more -- but it will surely bring customers into the fold for the industry as a whole. Each I-Track InfoGuide consists of two similarly configured files totaling 750 or more listings, one for contacts and one for institutions.

Release 1.0	 21 September 1987
Users can rename fields, including one that allows free-form text, but they cannot restructure them. Nor does adding a contact with his company add that company to the institution file; the two files are separate. Within these constraints, users are free to add data to I-Track files (adding comments or their favorite restaurants to I-Track's selections, say) or to create their own files. I-Track can import ASCII files, but beyond that its ability to interact with user's other software is limited. But there's one other great advantage: I-Track operates as a pop-up (with the data stored on disk), so that it can be used when it can be useful and stay out of way the rest of the time. That is, its benefits are great in proportion to its cost in time, bother, and resources.

GENERAL INFORMATION: INFORMATION IMPERATIVE

Far more powerful, rich, and extensible is release 2.0 of Hot Line from General Information, a new version that uses dBASE files and is far more flexible than the original (which sold 35,000 copies), runs on a network, handles dialing automatically, and includes 10,000 of the most frequently called listings, including names, numbers and addresses, for $75. Its files are far more extensive than I-Track's InfoGuides, but compressed to fit on a floppy. It will be updated at least annually -- but not necessarily with an "upgrade" discount. "Upgrades are for software," says Hot Line president Dick Brass. "Here, it's the data that makes the software worth having. After all, you pay full price for your new phone books."

General Information will shortly offer six extra data packs for under $50 with 10,000 listings each: government, travel, media & pr, computers & high-tech, and California (in conjunction with a telephone company).

Much the same approach, incidentally, is being taken by a number of other vendors, including Apple with its HyperCard. However, as befits a powerful, extensible tool, HyperCard's added values include applications and functions as well as just data. Apple also hopes to foster a large body of free and inexpensive HyperCard "stackware" from third parties. Other examples include IntelliCorp's SimKit, which customizes an expert system tool for building process-control and factory simulations, and the possible inclusion by Symmetry Corp. of a nationwide list of Domino's Pizza parlors with its Acta outliner for the college market.

REALITY TECHNOLOGIES: THE PRETTIEST SIGHT IS IN THE MIRROR

Another company whose marketing appeal lies in the data (even if they aren't always real) is Reality Technologies of Philadelphia. That company's first product, Business Simulator, was a fairly bland business modeling tool with scenarios that sold well, 30,000 copies at $70 to date, through Electronic Arts and Venture Magazine as well as direct. Its next venture, called
BusinessWeek's Business Advantage, takes the same models and fills them with data from BusinessWeek cover articles. They are selling in a $300-per-year subscription service (12 disks) that has already attracted 12,000 in its first six weeks of availability from Brøderbund and BusinessWeek itself.

Most exciting yet is a new $50 package tied in with What They Never Taught You at Harvard Business School, the book by Mark McCormack, also distributed by Electronic Arts. Far more real-world than the other two, this business "game" deals with corporate politics, negotiations, time management, customer relations -- the sorts of problems faced by real managers, as opposed to financial analysts and market strategists. Business Simulator was about making decisions; What They... is about implementing them. With character-based graphics (to keep the market broad), What They... doesn't show you pictures of the players, but it begins to feel pretty realistic (much like Infocom's text adventures), especially with a faster-than-real-time clock ticking away.

You're standing alone at the bar as potential clients mill around you. Should you wave to someone? Introduce yourself to the tall dark man in the golf blazer? Ask the man next to you whom he knows? Make small talk or talk business? It's all in the menus, but you have to make the choice from an array that doesn't include right or wrong answers (except for leaving as soon as you arrive). And don't forget to ask for each person's card (which automatically stores it in your computer Rolodex®) and say goodbye before you move on to fresher bait.

Another day, in the office, committed to a workload of 79 hours per week, you may decide to hire someone. Resumes are displayed for you to choose from. Pick carefully; if the skills don't match the work you are about to delegate, there may be problems later. (And the fellow with the gap in his record may prove to be less than trustworthy.)

Reality's programs are built with the company's own Prolog-based authoring tool, which includes a random generation capability so you can play the simulations over and over without encountering the same circumstances again. Although certain types of people and quotes turn up again and again, they have different names and backgrounds each time. Forget someone's name? Well, that's the point. Keeping track of people is confusing, and people who do it well frequently get the better of those who don't.

After making the scenarios more realistic, what's next? Working on the principle that the proper study of a company is itself, Reality is now selling company-specific versions of Business Advantage (without the politics!) to any company who cares to provide the appropriate background and pay a $10,000-plus fee. The fee covers Reality's costs for the typical two-week project; the gravy comes from volume sales of the model to company training groups and employees. Reality has already signed up DEC, Nynex and Upjohn.

Next step? Well, the proper study of an individual is himself or herself. How about a personal financial planner, but with the friendliness and flexibility Reality has learned over the last two years of building "games"?
Who gets the kids?

One of the more interesting jobs in the software business right now is Randy Komisar’s. He is the new lawyer-plus for Claris, the Apple software to-be-spun off, and he is negotiating the transfer of rights to Apple’s applications from Apple to Claris. But what are those rights? This has always been a tricky question, especially concerning Macintosh products. Apple didn’t much mind if third parties used the "Macintosh interface" to write software to run on Macs, but that doesn’t mean the company wants to sign over or repudiate those rights in an explicit contract with a third party.

Even within the company, there have always been divisions -- which must now be resolved -- over how Apple should define its proprietary interest in the interface and technology and how widely it should license it. In particular, should the interface be restricted to use on Macs only? "Fortunately," says Komisar, "even when I was at Apple I wasn’t the hawk on this stuff." Now, for the record, he believes that "it shouldn’t be a problem as long as Claris does the Macintosh implementation first."

The exercise is more than intellectual. Aside from two undisclosed agreements (with Digital Research and with Microsoft), Apple has never clearly outlined what it believes constitutes the proprietary parts of the Macintosh interface. It’s just like any other divorce, setting forth all those facts that you could gloss over as long as you stayed married.

If it can’t be fun, at least it should be quick

We last wrote about Intuit’s Quicken when it was a consumer-oriented checkbook program, designed not to help you manage your money better so much as to reduce the time it took you to do it at all. Now the company is releasing a new version designed to accommodate businesses more fully, with more flexible reporting, forecasting, and multiple accounts. It turned out, says Intuit founder Scott Cook, that lots of businesses were using the product just to write checks and keep track of them without the burden of double-entry accounting. These are not just small businesses, either, but branch offices of large businesses such as AT&T and PG&E. It’s a classic example of the using-1-2-3-for-wp syndrome: people will use what they can rather than what some designer thinks they should. Quicken is eminently usable, because Cook is a fanatic about designing products for users and makes his engineers sit in on focus groups. "They hate to see three weeks’ work trashed by a group of housewives," he notes, and they’ve gradually learned that "users don’t make mistakes; they have habits" that must be accommodated.

---

4 A job defined by Komisar’s former boss Al Eisenstat, one of Apple’s first grown-ups. Al joined Apple in 1980 as general counsel, and is now corporate secretary and senior vp with broad responsibility for strategy and administration.

5 Apple recently testified at a U.S. Copyright Office hearing in favor of separate copyrights for screen displays and program code, but without specifying what those might include.
Although Cook was formerly a brand manager at Procter & Gamble, Intuit sells Quicken with ads targeted at direct response. That is, the ads explain the product clearly enough to persuade someone to buy it, even including a little technology pitch to explain why Quicken is so quick (in the same way that a miracle ingredient explains the promise of plaque removal). Like Borland's ads, they do so effectively enough that Quicken sells well at retail, even though no salesman would waste his time pushing a $50 product.

RELEASE 2.0: REHASHED NEWS

Sybase gets sighted

Apple Computer has just invested an undisclosed amount (about $1.3 million) for an estimated 5 percent of Sybase, described in Release 1.0, 1 July 1986. Although it will take a year or so for Sybase to build a Mac version of its relational dbms, its presence (and promise) should significantly improve Apple's credibility with customers. Sybase uses SQL, the lingua franca of dbmsses these days. Sybase is a big exponent of the two-part workstation/server architecture. Although details are vague, we expect Sybase to implement the server portion of its software on the Mac under UNIX, while the end-user interface tools may operate in the native Macintosh environment.

Jef Raskin sets the record straight...

In our 12 May article on Canon's Cat, designed by Information Appliance ceo and Macintosh designer Jef Raskin, we noted the difference between the two products. Raskin writes: "I have been musing about [your commentary] for a while... I have not 'disavowed fancy interfaces and menus' [as we wrote] but originally wanted the Macintosh to be much simpler, easier, and faster to use than it turned out. My original Mac software design suffered badly from 'creeping elegance' and slavish borrowing of Xerox's work after I left Apple. Whether what I had originally designed would have been more successful will forever be an unanswerable question as history has moved on. I can tell you that everybody who has moved from a Mac to the Cat finds it most onerous to go back to using the Mac to do tasks for which the Cat was designed. I still use my 'millionth' Mac [given him by Apple] for CAD tasks, big spreadsheets, and music typography, but the Cat for everything else."

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 and connectivity markets, and the transformation of artificial intelligence into a commercial technology. Editor & publisher: Esther Dyson; associate publisher: Sylvia Franklin; circulation manager: Hyacinth Frederick; copy chief & consulting editor: William M. Kutik; secret source: Irene Nesbit. 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.
PHONE NUMBERS

Cliff Reid, Advanced Decision Systems, (415) 941-3912
Ted Durr, AIRS, (301) 454-2022
Tony Barrese, AT&T, (201) 522-6345
Bob Flast, American Express, (212) 640-4718
Marty Kahn, BRS Information Technologies, (212) 247-7770 or (800) 468-0908
John Clippinger, Brattle Research, (617) 492-1982
Randy Komisar, Claris, (408) 973-3120
Spike Kasper, Document Systems, (203) 846-1153
Jerry Kaplan, Robert Carr, GO Corp., (415) 775-5557 (due to change shortly)
Dick Brass, General Information, (206) 828-4777
Jef Raskin, Information Appliance, (415) 328-5160
John Gale, Information Workstation Group, (703) 548-4320
Scott Cook, Intuit, (415) 322-0534
Bill Spear, I-Track, (214) 578-8104
Gary Kildall, Tom Rolander, Chris Bowman, KnowledgeSet, (408) 375-2638
Robert Akscyn, Dan McCracken, Knowledge Systems, (412) 733-8333
Will Blanchard, Paul DeLorey, Lotus Development, (617) 577-8500
Frank Halasz, Jeff Conklin, MCC, (512) 338-3648, 343-0978
Mark Mackaman, Bill Gates, Microsoft, (206) 882-8080
Jim McGowan, Palladian Software, (617) 661-7171
Matt Koll, Personal Library Software, (301) 926-1402
Draper Kauffman, Quantum Access, (713) 622-3211
Mark Goldstein, Reality Technologies, (215) 387-6055
Bob Epstein, Sybase, (415) 548-4500
Randy Trigg, Richard Burton, Xerox PARC, (415) 494-4000

COMING SOON...

- Connectivity: Promises, promises.
- Agenda managers -- a new category.
- Groupware -- software from start-ups.
- Announcements from Lotus, Microsoft.
- Are expert systems intrinsically friendly?
- Nitty-gritty experts.
- The U. K. software market.
- And much more...

Release 1.0 21 September 1987
27

RELEASE 1.0 CALENDAR


October 1  Oracle at the analysts' - New York City. Sponsored by the New York Society of Security Analysts. Contact: Judy Zatz at (212) 344-8450.


October 4-8  Computer Services and Communications & Information Systems Seminars - Baltimore. Sponsored by Alex. Brown. The standard in Wall Street software events, combined with communications and information systems this year. Call Chris Mortensen, (301) 727-1700.

October 4-8  OOPSLA '87 - Orlando, FL. The second annual conference on object-oriented programming, sponsored by ACM and chaired by Adele Goldberg (ParcPlace) and Chet Wisinski (PPI). With Kurt Schmucker, Mike Nastos, others. Contact: Jerry Archibald, (914) 789-7695.


October 17-18  Softeach - Toronto. Training/pitching for a profusion of products, classroom style. Sponsored by Softsel. Call Laura Keezen at (800) 325-9189 or (314) 827-1724.

Release 1.0  21 September 1987
October 17-18  Computer Professionals for Social Responsibility annual meeting & banquet - Cambridge, MA. With keynote by economist Lester Thurow: "Does military spending hurt America's economic performance?" Contact: Gary Chapman, (415) 322-3778

October 19-21  Hammer Forum 87 - Cambridge, MA. Sponsored by Hammer & Co. With Bob Flast, American Express; Norman Lewis, Ford Motor; Bob Berland, IBM; Russell Harrison, American Airlines; and others. Contact: Pamela Davis, (617) 354-5555.


October 20  Fourth-generation languages - Burlington, MA. Membership meeting, sponsored by Massachusetts Software Council. Get together to hear the Yankee Group and talk about common problems. Contact: Joyce Plotkin, (617) 437-0600.

October 20-22  Networld - Dallas. Sponsored by Novell. Contact: Annie Zdinak, (800) 237-0316 or (201) 569-6916.

October 21  Venture capital/Initial public offerings seminar - New York City. Conducted by Accel Partners; sponsored by Adapso. With the voices of experience from Harry Reinstein (Aion) and Craig Hill (Cortex). Contact: Kelly Bailey, (703) 522-5055.

October 24-25  Softeach - London. Training/pitching for a profusion of products, classroom style. Sponsored by Softsel. Call Laura Keezen at (800) 325-9189 or (314) 827-1724.


Release 1.0  21 September 1987

October 28  Venture capital/Initial public offerings seminar - San Francisco. Conducted by Accel Partners; sponsored by Adapso. With the voices of experience from Larry Michaels (Santa Cruz Operation) and Kent Petzold (ViaSoft). Contact: Kelly Bailey, (703) 522-5055.


November 2-5  Electronic imaging '87 - Boston. Sponsored by Institute for Graphic Communication. Contact: Richard Murray, (617) 267-9425 (program) or Ed Martin, (617) 232-3976 (exhibits).


November 8-12  Data Training Conference - Anaheim, CA. Sponsored by Data Training Magazine (Weingarten Publications). Why Johnny can't compute... How to make documentation readable and products usable... For dp trainers and people whose products need them. Contact: Catherine Sununu, (617) 542-0146.


November 9-12  Autofact - Detroit. The event in factory automation. Sponsored by the Society of Manufacturing Engineers and other groups. With Richard Goatman (Rolls-Royce) Bill Poduska (Apollo, Stellar) and possibly Secretary of Defense Cap Weinberger. Contact Judy Evola at (313) 271-1080.

Release 1.0  21 September 1987
<table>
<thead>
<tr>
<th>Date</th>
<th>Event Description</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>November 13-15</td>
<td>Hypertext '87 - Chapel Hill, NC. Sponsored by IEEE, ACM, and other worthy groups. Contact: Frank Halasz, co-author of NoteCards, (512) 338-3648, or John Smith, (919) 962-1792.</td>
<td></td>
</tr>
<tr>
<td>November 15-19</td>
<td>IIA annual convention - Chicago. With Jim Treybig (Tandem), Phil Lemmons (Byte), Joseph Dionne (McGraw-Hill), and others. Sponsored by Information Industries Association. Contact: Mike Atkin at (202) 639-8262.</td>
<td></td>
</tr>
<tr>
<td>December 9-11 14-16</td>
<td>Data base 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.</td>
<td></td>
</tr>
<tr>
<td>December 14-16</td>
<td>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.</td>
<td></td>
</tr>
<tr>
<td>February 8-10</td>
<td>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.</td>
<td></td>
</tr>
<tr>
<td>February 21-24</td>
<td>ELEVENTH ANNUAL PERSONAL COMPUTER FORUM - Naples, FL. We moved it in search of variety and better weather. Registration forms will be mailed to subscribers this fall. For further information, please call Sylvia Franklin, (212) 758-3434.</td>
<td></td>
</tr>
</tbody>
</table>

Release 1.0 21 September 1987
February 25-27  Workshop on technology and cooperative work - Tucson, AZ. 
Sponsored by Bell Communications Research and the University of Arizona. Contact: Robert Kraut, (201) 829-4513 or Jolene Galegher, (602) 621-7477.

March 1-3  Third international CD ROM conference - Seattle. Sponsored by Microsoft. Contact: Sherri Eastman, (206) 867-3305.


March 14-18  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.

March 16-23  Hannover Fair CeBIT - Hanover, West Germany. Contact: Donna Peterson Hyland, Hannover Fairs USA, (609) 987-1202.


August 22-26  AAAI-88 - St. Paul, MN. The seventh annual. Sponsored by the American Association for Artificial Intelligence. Contact: Claudia Mazzetti, (415) 328-3123.

September 26-28  Second conference on computer-supported cooperative work - Portland, OR. Sponsored by ACM. Contact: Suzanne Sylvia, (617) 225-1860.

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

Release 1.0

21 September 1987
SUBSCRIPTION FORM

Please enter my subscription (at least 12 issues per year) 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.

Name __________________________________________
Title __________________________________________
Company ________________________________________
Address _________________________________________
City____________________ State _______ Zip__________
Telephone ____________________________

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
21 September 1987