

"Seven Issues": Revisited Hypertext '91 Closing Plenary

by Frank G. Halasz

Copyright © 1991 Xerox Corporation

[HOME](#) [SLIDE
PRESENTATION](#)

Thanks to Cindy Kunz of Texas A&M University for the following transcript.

Contents

- [Part 1: Introduction](#)
- [Part 2: "Seven Issues": Reconsidered](#)
- [Part 3: "Seven Issues": Reviewed](#)
- [Part 4: "Seven Issues": Renewed](#)
- [Part 5: Retraction and Conclusion](#)

It's nice to be the Zsa Zsa Gabor of hypertext. Those of you who've been to PARC will see that's even the name on my doorplate at PARC, thanks to Cathy Marshall.

I thought I'd take a minute out of my talk time to mark what I think is a very important passage in our field. As many of you know, sometime during the last year, I'm not even exactly sure when, the IRIS Project, the Institute for Information in Research and Scholarship at Brown University, I guess you call it an industry took a redirection or a tremendous downsizing. Essentially the Intermedia system stopped its growth in evolution at that point and passed into more or less history, although it's still running as we saw it here. And I think that's an important passage in our field and I'd just like to mark that occasion. I mean I think Intermedia, over the last seven or eight years, has introduced into our field a lot of ideas that we now take as commonplace. I think the most important one is the notion of anchors. I remember Norm Meyrowitz trying to convince me that was a good idea. Of course I didn't think so because NoteCards didn't have anchors but since then, in fact even then they were called blocks, not anchors. But I think as I walk around the Demos and hear the talk today, everyone's talking about anchors and that's an idea that I think is really due to the Intermedia project. The notion of a web of links as opposed to the collection of documents, the notion of a rhetoric of hypertext first introduced by George Landow at Hypertext '87, the very important word 'bazinga' which was repeated 100 times at the keynote last time and a whole host of other ideas that came from the Intermedia and IRIS project. So to Norm Meyrowitz, Nicole Yankelovich, Paul Kahn, George Landow, Karen Catlin, Tim Catlin, Bernie Haan, Victor Riley, and all the others who I don't know personally on the Intermedia project, I want to thank you for all the ideas. I want to thank you for the friendly competition over the last seven years. And I hope to see much more in all your new places. So I just want to thank you.

Introduction

Well, it's great to be here, deep in the heart of Texas, revisiting the Seven Issues. For those of you who aren't aware, the Seven Issues refers to the paper called "Reflection on NoteCards: Seven Issues for the Next Generation of Hypermedia Systems" which was presented at Hypertext '87 and then republished in the July '88 version of the Communications of the ACM. The paper is actually a little bit older. I consider this basically the fifth anniversary of

these ideas. Most of the ideas were part of a never completed, but unfortunately frequently cited NoteCards paper that Randy Trigg and Tom Moran and I were writing in '86. And then in '87 I wrote the Hypertext '87 submission and then in '88 I revised it for the CACM.

In the 87-88 time frame, I was living in Austin, Texas, where I was becoming quite a fan of country and western music and there was this great song on the radio whose lyrics went something like "happiness is Lubbock in the rear view mirror." And those of you who have been to Lubbock, Texas, know how true that is. But I began to think of this paper as "Happiness is NoteCards in the rear view mirror," that this was my last look backward at NoteCards and what I'd learned from it.

I thought an interesting way to start this talk, well actually I discovered this, this I guess is archaeology or something, as I was looking through the slides for my '87 talk, I discovered that this was the last slide from my '87 talk and I got severely dinged at the '87 conference for actually saying what I said here and so it never appeared in the printed version. So basically the last slide in the '87 talk, this is actually a photograph of that slide, claimed that the long term future of hypermedia lies in self obsolescence. Basically, I was trying to claim that hypermedia should disappear into the background of computing and that, in fact, once it disappeared in the background of computing, it shouldn't be an interesting topic anymore. Basically, I claimed that Hypertext '90 should be no more interesting than the '87 workshop on hierarchical file systems. Now, I never was at the hierarchical file system workshop so I don't know how interesting it was. I wasn't at the European Conference on Hypertext in 1990 but I read the proceedings carefully and it sure was a very interesting conference. And I hope my talk today will convince you that, in fact, maybe even if it is disappearing into the woodwork of computing, it remains a very vibrant and alive field.

The goal of the Seven Issues paper - basically I took NoteCards as a foil or as a backdrop, against which to develop first an assessment of where I thought the hypermedia field was, the state of the art of building hypermedia systems was in 1986 or 1987. And then from that I tried to propose an agenda for future research and development in hypermedia systems, kind of a proposal for what we as a field should look at as the core set of problems that we should be approaching in the next few years or, as I phrased it then, "in building the third generation of hypermedia systems."

My goal today is similar, or analagous. I'm going to use the Seven Issues paper as a backdrop or a foil against which to again assess the current state of the art in hypermedia. Notice I've crossed out the word 'systems.' Unlike in the '87 paper where I really tried to focus on the notion of hypermedia systems, I'd like to take a much broader perspective today and look at the larger world of hypermedia, systems included. Once I've assessed this current state of the art, I'd like to again propose an agenda for the future research and development in hypermedia, leaving off systems, to some extent, and propose this as a core agenda for what the field ought to be looking at in the next few years. Again, this is a proposal and I expect lots of argument with what I proposed in the question and answer session.

This is the plan for the talk. The first section is called "Seven Issues: Reconsidered." I'm going to kind of ask whether the lenses through which we were looking at the field in 1987 are still appropriate today. Then, Seven Issues: Reviewed, I'll take the seven issues I proposed in 1987 and ask how we've done with those. And, finally, I'll offer my new seven issues, which I'll tell you ahead of time, are actually ten issues for 1991 or 1992, as I've said there.

"Seven Issues": Reconsidered

Let's step right into the first section, Seven Issues: Reconsidered

The basic question I want to ask is if the eyeglasses through which we looked in the Seven Issues paper, are they still appropriate for looking at the hypermedia world today. I think that the Seven Issues paper, in retrospect, takes a very simple and narrow view of what the world of hypermedia encompasses, what was of interest to us as hypermedia researchers. Basically, the way we were looking at the world then was there was this thing called a hypermedia system. Basically it provided node and link storage, a set of tools we'll call browsers, and a few other tools for dealing with this basic node and link structure which you're storing away. On top of that maybe we'd put a few application veneers, little customizations to take this basic node and link model and make it appropriate for some tasks. And then there were people using this system. I guess we'd call them users and we were interested somewhat in the users and occasionally in multiple users using the same system. And this was basically the kind of objects in the world that we were interested in. The first, and I thought it was instructive, this is the first line of the CACM paper where I attempted to do a definition of hypermedia which I think very much fits in this view of the world. I said that hypermedia is a style of building systems for information representation and management around the network of multimedia nodes connected together by typed links. And I think that definition kind of describes the way, at least I was, and I think the community was looking at the world of hypermedia in '86 and '87. Well that was then. This is now. And I think, four hypertext conferences later, we've really changed what we consider to be the world of hypermedia and our world of study. Now in my talks I've been using this diagram instead to say what's of interest to the hypermedia world. I call this my flowerpot diagram and, since I had lots of clip art, I added a little rose to drive that home.

TAPE DROPOUT!

Here is a short essay question from one of John Leggett's final exams. Compare and contrast each of the following pairs, 15 minutes each: KMS and gIBIS; HyperCard and Xanadu; Trellis and SuperBook; and Ted Nelson and Rob Akscyn. I don't know about you, but especially if there's no partial credit, I would fail this exam.

The point of diversity was really driven home to me last year when I received in the mail, within about a month of each other, two books, which I both felt, in fact I ordered both of them. And I both felt they were indicative of my field. I read both of them and realized there couldn't be two books more different than these two. The first one is "HyperDocuments and How to Create Them" by James Martin. Those of you who don't know James Martin, he's written at least 50 books, and every one of the books is the epitome of practical MIS advice. I mean these books are for the MIS professional who wants to know something about the field. This is an easy read, takes about an hour to get through the whole book, and it's actually full of a lot of good advice. On the other hand is J. David Bolter's book called "Writing Space." It's a brilliant and, from a technoweanies point of view, a very literary book investigating the notion that electronic writing, or hypertext, is, in fact, the fourth age of writing following on the papyrus, the medieval codex, the book, and then finally the electronic writing. If you haven't read this book, I highly recommend it. It's a very enjoyable and interesting read and, again, especially if you're a technical person. The difference between these two books is astounding. Yet they both represent the field of hypermedia as it now stands and I think that's very interesting.

In my '87 paper, I actually tried to deal with the notion of diversity by proposing three dimensions through which we ought to look at the field of hypermedia and classify what we're talking about along those three dimensions because where you are on those three dimensions has a lot to do with what you think the important issues of the day are. The first dimension I suggested was scope. You know there are hypertexts that run on little palmtop computers that you carry around in your coat pocket. On the other hand, at the other end of the scale, there's public access Xanada, which is Ted Nelson's proposal that on every corner there ought to be an information McDonald's, and all the world's literature would be available on line through the information McDonald's. Of course, in between these two extremes, there's single user workstation systems, there's workgroup hypertext on a LAN, there's enterprise wide hypertext running on a wide area network, and so on. The issues that arise in palmtop computers seem to me a little bit different than the issues that arise in public access Xanada. So if you're going to assess a system, you have to know where it stands on that dimension.

The second dimension I talked about was browsing versus authoring or, as the field has come to call it, and I hate these terms because they're not correct, but static versus dynamic hypertext. To me a dynamic hypertext has video in it, it's not one that changes frequently. But at any rate, what I was trying to get at is the difference between systems that emphasize the presentation of information versus systems that emphasize the creation and manipulation of network or other data structures. And there's a world of a difference if you're building a system for people just to read versus if you're building a system for people to write. Now a lot of systems fall in between that but I think it's an important dimension to pay attention to.

And, finally, I suggested there was an important dimension of task-specificity. There are a lot of generic systems in the world. Let's take Intermedia, to some extent, KMS, which are general hypertext systems. And there are other systems which are really designed for very specific tasks. gIBIS, Aquanet, the Concordia Document Examiner, are good examples of systems which had a very specific task in mind. And, again, you're going to run across different issues, depending on whether you're building a generic system or a system for specific tasks.

So those are the three dimensions that I proposed in 1987. Since I got rid of the Seven Issues I had to have another seven. I'd like to add four new dimensions that we need to think about and I think are very operative in this conference. It's been very interesting to see the diversity of papers presented. So the four dimensions I'd like to suggest this time, or four additional dimensions I'd like to suggest this time, I've made them, I guess they're made ad hominem, I've classified people by the kind of systems they build.

Let me go through the four new classifications here. First of all the navigators versus the architects. Actually, John B. Smith and his talk on ABC give a very similar distinction, for those of you who were there, between document centric frame based and graph centric systems. But in my classification, the navigators, their focus in the hypermedia system is on the nodes and their contents. And basically the standard activity on the navigation-based system is to navigate from node to node through the hypermedia space. Typical examples are KMS and HyperTies are prototypical examples. And this is what a typical navigator screen looks like. The whole screen, the whole user interface, is taken up by the node or the nodes. And the links on those nodes then you can follow to the other nodes.

I'd like to contrast that with the architects. To the architects, the focus is on the overall structure of the hypertext. We're interested in building structures and not necessarily in nodes. There are nodes but they're secondary. The standard activity in an architect's hypertext system is manipulating the structure of the hypertext using a browser or some sort of overview and not in creating and reading nodes. Typical examples of that, well the best example that I know, is our Aquanet System, or the gIBIS system from Conklin and Begeman. This is a typical screen from an architect-based system. The whole screen or most of the screen is taken up by the overall structure that you're dealing with and over in the corner here is the little node that you might be working with.

I think it's very clear that the kind of issues, the information management issues, that you deal with in a navigation system are different from the kind of, let's say, structuring and argumentation issues that you might deal with in an architect-based system. They're both hypertext but they take different views of the world.

Another distinction that I think is very prevalent in our field right now is the literalists versus the virtualists. The literalists, the structures (links) are created and represented explicitly in the hypertext. And you navigate by following these explicit structural connections from node to node through the hypertext. I think all the classical hypertext systems, Intermedia, NoteCards, KMS, are literalist systems. Aquanet, a system we talked about yesterday, is also a literalist system. Contrast that with the virtualists' systems where any structure is implicit in the form and content of the nodes and basically the system computes a structure over those nodes in order to aid the user in navigating through the information space or interacting with the hypertext. I think the SuperBook example that Dennis Egan talked about yesterday is a prototypical example of a virtualist system as is the Cybermap paper that Peter Gloor presented earlier in the conference. There are lots more examples. Frank Tompa's tutorial on Sunday was actually, I think, one long plea for virtualist systems in hypertext. But again, the literalists are going to think about the problems of hypertext very different than the virtualists.

In 1987, Jeff Raskin gave a brilliant talk in which he argued, I think, essentially against hypertext. He introduced this notion of the Card Sharks and these terms, the Card Sharks versus the Holy Scrollers. Unfortunately, he never wrote this up so the terms Card Shark and Holy Scrollers have disappeared from the literature. But I love the term. I wish I could be as clever. Basically, to the Card Sharks, each node in a hypertext is a fixed-size card onto which you attach items like text or graphics or links at some location on that card. And then again you navigate through the hypertext by jumping from card to card. Again, prototypical examples are KMS and HyperCard.

In contrast Jeff proposed the notion of the Holy Scrollers. To a Holy Scroller the hypertext is this collection of potentially very lengthy documents. And instead of jumping from card to card, what you really do is navigate by sort of flying through the document space, both within documents. You never navigate within a card but you navigate within a document and you navigate between documents. In addition you also use the standard method of scrolling through a document. So Doug Engelbart's system, Augment, has always been my example of a prototypical Holy Scroller. The FRESS system from Intermedia, from Brown University, shown on a videotape last night and the SuperBook are all examples of the kind of document-oriented as opposed to card-oriented systems. Again, my standard refrain, there will be very different issues raised whether you're a Holy Scroller or a Card Shark.

And finally I already eluded to a distinction in showing the Martin book versus the Bolter book. And I've classified this distinction as the Literati versus the Engineers. I looked very hard for the opposite term for Literati and I couldn't find it. I'm not really satisfied with the notion of Engineers. But to the Literati, the hypermedia is a new writing space. In contrast to the Engineer, hypermedia is an information tool. And I've tried to capture this distinction by two quotes. First, one from the Bolter book where Bolter claims that "this new medium is the fourth great technique of writing that will take its place beside the papyrus roll, the medieval codex, and the printed book. Hypertext will lead to a new definition of unity in writing." Quite a lofty goal.

In contrast, this is a quote taken from Bob Glushko's paper in the document processing conference in '88. "We created a compact disc version of the Compendium in which hypertext features enhance and supplement the table of contents, index, etc., as methods for finding information." Well you can kind of capture a different tone in the way they approach hypertext in these two quotes. That's very clear in that the interactive fiction-type papers in this conference were very different from the people who were writing tech manuals. I think they're both very important approaches but we have to be aware that we're looking at the world somewhat differently.

So diversity in our field is very important as is the notion of kind of a common definition of hypertext. Let me return to the question of "are the eyeglasses through which Seven Issues looks at hypermedia still appropriate?" Well, I think, in general, the Seven Issues paper wasn't entirely wrong in the eyeglasses it used but it certainly wouldn't hurt to modernize our eyeglasses a little bit. My clip art didn't turn out so good but those glasses had very nice flowers on the edge which makes them look a lot more modern than the black spectacles in the original talk.

"Seven Issues": Reviewed

Let me move on to the Seven Issues: Reviewed. These are the seven issues from my original paper and what I'd like to do with each of these seven issues. First of all, I'll try to capture for you very briefly how the issue was described in the Seven Issues paper. I'll talk a little bit about what I think the progress has been in the last four or five years on this issue. And then I'll say, 'But maybe I was wrong in 1987 in thinking about this issue and we ought to think about it differently today.'

So let me start with the first issue. The first issue was search and query. In my paper I said navigational access by itself is not sufficient. Effective access to information stored in a hypermedia network requires query-based access to complement navigation. And basically, in the paper I tried to make a distinction between two kinds of searching in hypermedia.

First of all, content search. That is where you search over all the nodes and potentially over all the links individually and look for nodes and links that match a specific pattern. The content of those nodes and links should match a specific pattern.

In contrast I tried to talk also about structure search where you take the network, at that time I guess, the network as a structure and you look for substructure, subpatterns. So find all nodes that are connected to another node by such-and-such a link. So you are looking over the structure of the network and not the content of the nodes. I truly believed in 1987 that structure search was the breaking edge of the field in hypertext and there'd be a floodgate of research on that topic in the next few years.

Well, has there been progress in the search and query domain? I think in the terms of content search there's been a tremendous amount of activity. In the 1989 hypertext conference, there were two full sessions dedicated to information retrieval and hypertext. Last year's ECHT conference, there was one session. In 1989, we saw a demo of full-text search in the Intermedia system. Recently I saw a really nice product called Dynatext from Electronic Book Technologies which has done a nice integration of information retrieval or full-text search over the nodes and hypertext linking. So I think there's been a lot of activity in the area of content-based search.

To my surprise, the notion of structure search really hasn't taken off. There's been a little activity. In particular the GraphLog paper that Consens and Mendelzon presented at the last American conference was a really nice attempt at doing structure search in hypertext. But in general the idea hasn't taken off. And people aren't really working on the problem. Perhaps it's a tough problem or perhaps I was wrong in assuming that it was really necessary for the hypertext field but that's still an open issue.

But, let me read to you this down here is the quote that I just read to you from the paper and this is the immediately preceding paragraph from the paper. I said that "In some sense, hypermedia is navigational access." Something I still agree with. Then I said, "The ability to browse around a network by following the links from node to node is a defining feature of hypermedia." Wrong. I don't think that's true anymore. In particular, I've been very influenced by the SuperBook system. SuperBook is a hypertext system. I remember when the SuperBook paper was presented in Hypertext '87 the title of the paper said, I mean basically it was something to the extent "Is SuperBook a hypertext system?" And they were very worried about whether SuperBook was a hypertext system or not. To tell you the truth, I think a lot of people were concerned about defining hypertext at that point and whether SuperBook was or wasn't. At this point you know I feel, more or less, who cares? SuperBook is a very interesting system so let's include it in the realm of hypertext systems.

In SuperBook there are no links and navigation is essentially done by using full-text search to navigate through this document space in the system. Very interesting in 1987, Jeff Raskin showed a really nice little kind of laptop computer called a Canon Cat which the whole computer, the whole operating system, was built around the same notion as SuperBook - that all you did was do full-text search to jump around the file system. I never found out what happened to the Canon Cat. It didn't make its appearance in the commercial domain as far as I know.

But I think the lesson we want to learn from both SuperBook and from the Canon Cat and from a whole lot of other systems is that links really aren't necessary for hypertext. That navigation by query should be considered a true alternative. Remember in 1987 we were calling it a complement to links. Now I'd like to call it an alternative to links, a true alternative to links, in designing hypertext systems. Of course, this leads to all kinds of interesting issues like what is a link anyway. Maybe we'll just call queries dynamic links and then we'll fit in the old definition. But I think that's beside the point. I mean I think we need to change our focus of the field to really accept this notion of query-based hypertext systems, or search-based hypertext systems.

Let's turn now to Issue 2 from the '87 paper. Basically that was on composites which I subtitled Augmenting the Node/Link Model. And I claimed that the basic hypermedia model

lacked a composition mechanism, that is, a way of representing and dealing with groups of nodes and links as unique entities separate from their individual components. And I claimed that all aspects of the hypermedia should support inclusion, that is, part-of, relations as a construct distinct from the standard kind of reference links we've been working with.

Has there been progress in this field? Certainly in the theoretical work. People who are building models of hypertext systems have really accepted composites as kind of a standard part of the hypermedia model. Of course that's easy for them to say because all they're doing is designing models and not building systems. But, for example, we heard the paper "The Nested Context Model" in this conference which really took the notion of composites very seriously. For another example, the Dexter Model that Mayer Schwartz and I and the Dexter Group worked on that takes composites as a first class object. In contrast, composites are beginning to appear in systems that people are building. Two good examples are the ABC System with the graph server that John Smith talked about in this conference and the HyperBase system that Norbert Streitz talked about at the ECHT conference.

It's interesting as I began to look for systems that included composites in their implementation, I began to realize there weren't a whole lot of systems around to look at. And this led me to an interesting sub-issue that I'm really not going to talk about today but where are all those third generation systems that were supposed to solve all these problems that I put forward. There's actually, in our field, compared to 1987, relatively little large-scale system building activity going on and I think that's an interesting problem. But not one which I'm going to address right now.

But what was wrong with the notion of composites. In re-reading the paper I began to wonder why I was limiting myself to the notion of augmenting the node and link model and not necessarily replacing it. I mean there are alternative data models for explicitly representing the relationship among nodes and several of them have actually been explored. The Petri Net model that Stotts and Furuta have put in their Trellis system and published, I think, in every recent conference except this one. I find that a brilliant piece of work and one of the really nice ideas to come into this field. It's still a network model but it's a different kind of network model. The Hypersets idea that Van Parunak presented yesterday, our own Aquanet work, are all examples which have more or less thrown out the notion of node and link model and taken another data structure as the basis for the hypertext. Just to drive that home, here's a picture of our Aquanet system that you probably saw yesterday. And this thing right here which we're calling a relation and has actually three parts and no really links to it whatsoever is the way we're thinking about the hypertext connections. There are still links as you see up here in our system or things that look like links but basically we've replaced links with the notion of a structured relation. And I think we'll see in the near future more and more people moving away from the strict node and link model and doing things like Hypersets and Aquanet and Trellis.

So let me move on to Issue 3 which I called Virtual Structures and subtitled Dealing with Change. Basically I claimed the hypermedia model needs to be augmented with a notion of virtual or dynamically-determined structures. And then I really narrowed the definition after that. And I said "virtual structures are defined by specifying a description of their components that is reified by a search procedure whenever the structure is accessed or instantiated. Basically you do a query every time you access the node or link and it finds the results and that's the node you look at. And I talked about virtual nodes, virtual links, and virtual composites in the paper.

Well has there been any progress in this notion? Well as I narrowly defined it in that paper, I could find only one paper that had an implementation of virtual structures and that's the PHIDIAS work of Ray McCall which was actually presented at the last conference and several other places. I also found one theoretical mention which was the notion of retrieval links in Steve DeRose's paper from the last American conference. In addition, technically speaking, the links in HyperCard are virtual according to the definition that I gave. But it probably is seldom virtual in intent. That is, most HyperCard card stacks have the command 'GOTO card 1675' as opposed to the kind of virtual command or the kind of query thing, GOTO the first card that you find by searching for this word. In intent, this is really a virtual link, whereas this is a hard-wired link that just happens to have a funny implementation through a script language.

Well, so there hasn't been a whole lot of progress as I saw the problem in '87. But I think I was way too narrow in my definition of what virtual structures were in the '87 paper. And I think there's been lots of activity in very related fields that we need to kind of include under this notion of virtual structure.

First of all, there's the notion of implicit links. Basically, links where you have a word in the hypertext and you can select it and then jump to an entry in the dictionary, that is, the links really aren't hard-coded in. But you can still jump around the hypertext by using things like Intermedia's Interlex and the kind of implicit links that appear in the Perseus system.

Also the notion of contextual links, links that appear or disappear depending on the state of the system or the state of the user at any given time. We heard a really nice paper by Guy Boy on links in his CID system where he talked all about contextual links. Similarly, we heard a very nice example in the paper on the coherency toolkit from the GMD-IPSI paper yesterday. The one called "Eliza in the Chinese Room" had some very nice examples of contextual links.

Other issues include discovered structures, that is, links or composites that are created by the system on the basis of computing a similarity among nodes. The CyberMap paper at this conference and the Link Apprentice from Bernstein are good examples of that.

Also the notion of computational nodes, nodes that when accessed go outside the system, do a computation, and bring the results in. A good example of that is the DSS, the Decision Support System Shell, that Michael Bieber presented at this conference.

But all of these things, implicit links, contextual links, discovered structures, computational nodes, are examples of virtuality in a hypertext system and I think we ought to include, if I were to rewrite the paper, I would definitely include this under the notion of virtual structure. So, in fact, there's been a lot of activity in this field in the last five years dealing with virtual structure if you broaden the definition a bit.

Let me move on to Issue 4 which is computation in and over hypermedia networks. And basically I claimed the hypermedia systems in the next generation, that is the hypermedia systems we see at this conference today, should accommodate the integration of hypermedia with computation. And basically I proposed two models of computation, one which was basically computation over a hypermedia system where there was a hypermedia system and then there was a separate computational engine, say an inference engine or something like

that which entered data into the hypermedia and read data back from the hypermedia system, much as a human user might use the same system, although probably through some sort of API.

In contrast, there was the notion of computational hypermedia where the computation and the hypermedia were actually merged into the same system and we had this notion of computational hypermedia. Of course I had no notion of what that would really be like but it seemed like an interesting idea at the time. If we look at the progress on that in recent times, I actually expected computation over hypermedia systems to be the dominant paradigm and computation in hypermedia systems as a natural part of the computation to be relatively rare. In fact, I think it's turned out the other way. Computation in hypermedia systems actually attracted a lot of attention. Once again, the paradigm example of that is the Trellis work that's based on using the Petri Net computational model as a basis for hypertext. Lots of other examples, one we heard at this conference was the WebTalk work of Nanard and Nanard. But lots of other people are working on the notion of putting computation into hypertext. Less work on computation over hypermedia systems but there's still some. Again let me just point to one of them, the Decision Support System Shell work that Michael Bieber presented at this conference. So in fact computation in hypermedia has made a fair amount of progress over the last four years.

But in reviewing the paper it became clear to me that at least I was mixing up two different ideas in the notion of computation. And basically I merged in the paper the notion of computation over hypermedia systems and the notion of knowledge representation in hypermedia systems. I was taking a very AI approach to the world and, as Danny Bobrow said, I mixed up inference with computation in the hypermedia systems. And I believe today that knowledge representation and hypermedia should be taken out as an entirely separate issue. Maybe not one of the top ten but it should be a separate issue from the notion of computation in a hypermedia system. And I think the issue of knowledge representation in hypermedia should be a separate issue and, in fact, it's generated a lot of interest. There were, in fact, three papers in this conference dealing with the problem of knowledge representation in hypertext. So if I were to rewrite this paper I would have included another issue that dealt with knowledge representation.

Moving on to Issue 5 which was versioning. I claimed that versioning is an important feature in hypermedia systems. A good versioning system will allow users to maintain and manipulate a history of changes to their networks. It will also allow users to simultaneously explore several alternative configurations. And I said something which I still believe today. The goal is to adopt and improve on the versioning mechanism that appeared in the PIE system. PIE was the system that Danny Bobrow and Ira Goldstein did way back in the '70s that was a hypermedia-like, software engineering environment. Brilliant versioning model in there and I suggested that the hypermedia field ought to follow up on that.

Well what progress has there been? Basically, none. I could find no published work on versioning in hypermedia systems after Mayer Schwartz and Norm Delisle's Contexts work in the CSCW '86 conference. And I could find only one unpublished manuscript on versioning that somebody had sent to me in my files. Let me add a footnote here that, in fact, there's a poster by Anja Weber in the posters session here dealing with versioning in hypertext so I'd have to correct this a little bit.

But by the way, side note here. I decided to do my 35mm slides, rather than like I should

have on a live demonstration with a computer - a hypertext version of these slides. And I'm very glad that I decided to do that because I think I would have spent the entire conference updating my talk based on what I heard here. And by having 35mm slides I didn't have to stay up all last night doing that. And this is a good example of that.

I'd also like to note there's a considerable body of work in versioning and related areas, things like CAD databases and CASE databases, but the work hasn't really been adopted in the hypermedia field. Again I went clip art wild late one night and put a little 'overdue' stamp here to indicate not too much work. But I began to think about it. Maybe versioning is just not a critical requirement for hypermedia applications. I think it's very interesting that, in this conference, the paper from Boeing on Industrial Strength Hypermedia where they basically laid out the requirements for industrial strength hypermedia, did not include versioning as one of those requirements. It included configuration management but the definition of configuration management did not include versioning. So maybe, I began to think, that, in fact, it wasn't important. During this conference, of course, David Gunning and the panel that Bob Glushko arranged, speaking from the Air Force's point of view, said that versioning in technical manual applications was, in fact, a very significant problem of theirs. So maybe, in fact, I have to revise this. I also began to think maybe that we've all lived with Unix and MS-DOS and MacOS so long that we just don't know what we're missing and that these systems that we've been using have destroyed our view of the world. But whether, in fact, versioning is important or not is still, I think, an open issue.

Let me move on to Issue 6, the support for collaborative work. I basically claimed that the next generation of hypermedia needs to focus on improvements for technologies for shared databases as well as improving the support for social interactions dealing with a shared hypertext network. So I really talk about two different issues under one issue. The system issues, which I called the mechanics of multi-user access, things like very long-term transactions, concurrency control, notification, some of the issues we talked about in the panel just before this. And I also talked about this notion of supporting social interactions over a hypermedia network in which I talked about two issues, mutual intelligibility, that is, can I understand what you're doing in the network if we're collaborating over that network. And the notion of rhetorics for hypermedia I also included under this notion support for social interactions.

Well, has there been any progress? In the system-level support, in fact, I think there's been some amount of progress in this area. I think multi-user back-ends are becoming kind of a standard hypermedia facility. When you build a system, you tend to build a back-end that has multi-user facilities. Mostly people are building them on existing relational databases like Sybase and all they're doing is adopting the multi-user capabilities of that relational database system, the kind of short-term transactions, some minimal notification facilities, and so on. But they really aren't going beyond what the standard engines are providing. But they are doing the minimum. I don't even know how the graph server is implemented but Hyperbase is a good example of that. Originally I thought the Virtual Notebook System was a similar example. Although in a recent talk I was very impressed with how much the Virtual Notebook System has taken some hypermedia-specific collaborative features into their back-end database, things like asynchronous notification of changes, and so on.

Turning now to the support for social interaction. This notion of mutual intelligibility, which actually was first introduced by Trigg and Suchman in their paper on hypertext at CSCW '86, really that idea has disappeared from the field as far as I could tell. And I think it's something

we might want to pick up again.

This notion of rhetorics for hypermedia is, in fact, a hot topic. So hot that my font turned orange. I included the notion of rhetorics for hypermedia under collaborative work in '87 but when I re-read the paper I said 'why the heck did I do that?' It's very clear that the notion of rhetorics for hypermedia should be an issue of its own. A very short description of the issue is how to write readable hypertexts. And again there is a very closely related issue called hypertext engineering. I don't really want to make a distinction between rhetorics and hypertext engineering. There's a tremendous amount of interest in this. There are four papers which I classified in this conference on that topic. There are probably another five or six papers in the '89 and '90 conference which I just didn't include here. So it's a very hot topic and something I think will generate a lot of interest.

Another thought that I had on the issue of collaborative work was that, in my '87 paper, I didn't include the notion of facilities to support very specific collaborative processes as part of the larger CW issue. I talked about multi-user back-ends and multi-user facilities but not process support. And I think there's been a lot of interest in that area in the field in the last four years. I think the best example of that may be the annotative collaboration work, the InterNote work, that was presented at the last American conference, where people were looking at draft passing and building specific tools to support the collaboration through draft passing. Another good example, of course, is Chris Neuwirth's work on PREP where she's looking at coauthoring. But these are systems that built very specific collaborative processes in there and I think that's something we ought to pay attention to.

Turning now to the last issue from the original paper, tailorability and extensibility. I basically claimed that hypertext is far too generic and that most of the systems, to get over that, had built-in tailoring facilities to allow people to modify their system to their particular needs. But the tailoring that had been built at that time, the tailoring facilities, were really aimed at system implementors and programmers and that our goal, or our challenge, was to bring tailorability into the hands of the non-programming user. And I still believe this goal today although I get pounced on every time I say this, I think our goal is to match GNU Emacs in terms of its tailorability and extensibility. And those of you who love GNU Emacs will agree with me. Those of you who think it's a curse on the world, you can eliminate that sentence.

Has there been any progress? Well, in fact, there's been a fair amount of progress. The notion of scripting languages. Interesting in '87, HyperCard had not yet been announced. Since then, of course, HyperCard and the notion of scripting languages has taken over so that you can tailor the system by using the internal scripting language. I question, however, whether scripting languages are really for non-programmers. It takes a fair amount of expertise to really effectively use a scripting language in most of these systems.

The notion of schemas and templates has appeared in the field in the last four years. Yet in this conference there were three papers which talked about this idea but the idea is that you build a sub-network that kind of becomes a standard rhetorical move in the system and that these are the kind of things that users can create and then propagate to other users in order to tailor the system.

And, again, the notion of paths or guided tours of which there were two very nice papers in the last American conference which allow people to suggest to other users the way they ought to navigate through the system.

These are all examples of how users can tailor the system to meet their own need. And I think there's been progress and there will be more progress in this area because it's very important. On the other hand, I think our notion of tailorability is going to change in the near future. And basically, with the advent of open hypermedia systems, which I'll talk about a little bit later in the talk, the responsibility for tailorability and extensibility, I think, is going to change. Today, to some extent and certainly in '87 and '88, we thought about the world this way: there was this thing called a hypermedia system, and as part of that we built this scripting facility that was unique to the hypermedia system. We don't build hypermedia systems this way anymore. At least, we shouldn't. We should be building them like this - that there's a set of interacting processes that are responsible for different parts, the user interface, the storage, the link service and the scripting facilities as just part of that. In fact, this shouldn't be hypermedia-specific but it should be just the scripting facility for the whole open system. And so I think what we'll see is just a change in the way we think about tailorability. Tailorability will become a more generic problem of systems and not a hypermedia-specific problem.

So let me summarize my review of the Seven Issues. I think a lot progress has been made but much, much more can be done on the original seven issues as I thought about it. But I think we need to revise our agenda to match changes in and around the world of hypermedia over the last few years.

"Seven Issues": Renewed

And that leads me to the last section of my talk which is Seven Issues: Renewed. What are the major issues facing the hypermedia community in 1991 as opposed to 1988. As I said, instead of seven I have 10. I have kept the magical number seven under the technology issues but I've added a new category. All of the original seven were basically technology issues but I felt it very important in 1991 to talk about a whole other set of issues which I have defined as market issues. In fact, a year or so ago, Bob Glushko came to PARC and gave a talk called the "Real Seven Issues in Hypermedia Systems" and, in fact, they were all market issues and I basically accept that point of view but there aren't really seven, there are only three.

Turning first to the technology issues, these are the seven new issues and I just want to trace what happened to the old seven issues. First of all, I took the first three issues, search and query, composites and virtual structures, and basically compressed them into one issue called 'Ending the Tyranny of the Link'. I took the notion of computation and preserved it. I guess I moved it down on the list. I took the notion of collaborative work. I preserved it. I took the notion of tailorability and I preserved it. I eliminated versioning from the top seven technological issues, not that I don't think it's important but I think there are other issues at this point that are more important. And then I added some new issues: open systems, user interfaces for large information spaces, and the problems of very large hypertexts. So today I won't cover all seven issues because I've covered several of them. I'll cover the ones that you probably can't see but they're outlined in blue, the ones that are basically new to my list of seven issues. So let me move on quickly to that.

Ending the Tyranny of the Link. I guess the title is more fearsome than the message. Basically I would like to claim we need to develop a new conception of hypermedia that includes non-network structures as well as virtual structures on an equal footing with network structures. If you've heard the rest of my talk that should be no surprise that I'm proposing this now. And I think that this will take hypermedia into some new directions. They're not really so new because people have been exploring them. But as I said, purely navigational

systems based on content search will become much more common and SuperBook is my example of a prototype. If you look at systems like Augment and FRESS, one of the feelings that you get is that you're flying through texts. And that notion of text-flying has been lost, I think, in our field. And I think systems like SuperBook and some of these future navigational systems will get us back to this notion of text-flying which I think was really nice in Augment and FRESS and we've lost it somehow.

We'll also see what I've called highly structured supra-network hypertexts, things like our Aquanet that we showed yesterday but also things like Hyperset that Van Parunak presented. By supra I meant other than network structures. And finally, we'll see hypertext that have inherently computational models built into them. And again Trellis with the Petri Net model is my example of a prototype of that direction. So I think we're going to see a whole lot of different kinds of things we call hypertext other than this boring, old directed acyclic graph that we've been dealing with for the last few years. I think allowing for a broader array of data models in hypertext will really improve our ability to interact with different complementary systems and technologies and, in fact, in solving more problems than we've been able to solve. So I think the integration of information retrieval and hypertext, it's always been kind of a funny, never quite sure how to get the two together, will become commonplace as we look at these systems like SuperBook. Similarly, knowledge representation systems, expert systems, CAD systems, they'll all fit better with hypertext if we allow hypertext to be a little broader than we have in the past few years. So I think our goal for the next few years really is to kind of take on, not to kill off, the link. It's still a very important concept, but we need to allow these other types of systems to come in.

Let me turn to technology Issue 2, that's open systems. It's very clear that the hypermedia system of the 1980s, the monolithic hypermedia system, things like NoteCards and even Intermedia, is no longer viable. They're going to be replaced by open hypermedia systems in which you have independent but communicating components which, taken together, produce hypermedia functionality. Amy Pearl gave a nice description of this, I think, in her '89 paper on the Sun Link service. Doug Engelbart argued for it in his open hyperdocument system in CSCW '90. I think the issue we have to face is, what is the appropriate decomposition of hypermedia functionality into independent components? How do you take this thing we call hypermedia and divide it up into separable pieces. I think it's very interesting seeing the Linkway product in the demonstrations from DEC. In my mind they've done a nice job of kind of building a link service as an independent component but they made an interesting error in putting the anchors with the links instead of with the nodes. Maybe I'm wrong about that but that's my intuition that they've done the wrong decomposition and this is something that we have to face as we come to open hyperdoc systems.

Similarly, once we've got it divided into components, the problem is what are the communication protocols by which these independent components will coordinate themselves. And let me give a forward link to my market issue on standards in this regard. I think a very important challenge in open systems will be taking advantage of what is a rapidly evolving field of standardized mechanisms for object-oriented, inter-application communication facilities. Every system vendor in the world, at least that's going to survive in the next few years, is offering a way to knit applications together in their system. Things like MicroSoft's OLE, HP's New Wave, Apple Events, the Object Request Broker from the Object Management Group, the distributed object management facility from Sun and HP and on and on and on. These are infrastructures for taking applications and linking them together, maybe not linking as we think about them, but linking them together as systems builders think of

that. And I think that much of the functionality we now build into our hypermedia systems is going to be provided by this infrastructure that the systems vendors are providing. And our challenge in the hypermedia community is to figure out how to take advantage of OLE or Apple Events in building open hypermedia systems.

Let me turn now to the third issue, Issue 4, and that is user interfaces for large information spaces. Basically, for a long time we've had a problem with building an interface that allows a user to manipulate a very large network or other structure on a workstation screen. Basically, in NoteCards we had browsers that got up to 500 nodes and that was a big deal because you couldn't deal with it, it took 10 minutes to redraw and all this other kind of stuff. In Aquanet we have browsers that are required to show 1,500 objects on the screen and they actually have the user manipulate them. If you haven't seen our demo and watched it take two minutes to draw this, this shows about 750 objects on the screen. It doesn't look like that because they are all stacked one on top of another. But building an interface that the user can actually manipulate each one of these 750 objects and change the structure actually turns out to be a big challenge. And, in fact, Aquanet has not solved that challenge.

There are a lot of solutions, or partial solutions, to this problem of interfaces for large information spaces. Fisheye views that George Furnas introduced is an interesting example. Three-D rotating graphs, nested set displays that Steve Feiner built into his IGD hypertext system, a really nice one that one of Ben Shneiderman's students, Brian Johnson, is working on is called Treemaps, should be coming out somewhere soon. Very interesting idea of just how you might build a radically different view of a network structure. And I think this issue is likely to grow in importance in the near future as we build larger and larger hypertexts and hypertexts become prevalent, we're going to have to face this problem of how to build an interface for these large systems.

Let me move on to Issue 5, the fourth new issue, the problem of very large hypertexts. I mean it's always been one of the main selling points, I mean this is Ted Nelson's selling point, the hypertext vision, that it's proficient for handling very large hypertexts or very large document collections. And by very large, we mean 10,000 or 100,000 or a million documents. Unfortunately the reality has yet to catch up with this vision. In this conference, if you read the brochure, the program, there's a discussion of the ACM Hypertext Compendium, which is a very nice piece of work, but it's subtitled something like dealing with large hypertexts. Well I don't know how big the Hypertext Compendium is but my guess it's about 500 documents or maybe 1,000, if we're lucky, documents and that's not a very large hypertext. That's kind of a teeny hypertext. You contrast that with the Boeing paper on requirements. If I can remember, the 777 airplane will have four terabytes of information to deal with and how do you build a hypertext system for four terabytes of information, not the ACM compendium, is the problem of very large hypertexts. And I think our challenge in the next generation, I think has been repeated throughout this conference by anybody who comes from industry, is to build useable, industrial-strength hypertext systems capable of handling 10,000 or 100,000 documents.

And this is going to raise all kinds of interesting issues, some of which have purely to do with scale and building very large systems, some which have to do with the problem of disorientation in very large information spaces. But interestingly we'll get all kinds of what should be kind of side-like problems, I think. There was someone on a panel from Silicon Graphics whose name escapes me right now who talked about heterogeneity. If you're going to have 100,000 documents, they're going to come from 37 different word processor formats.

How are you going to handle that problem when you build these large document spaces? So you're going to have to deal with all these interesting problems that are kind of related to the problem of building very large hypertexts. But again the challenge is to build large hypertexts, not to stay in these tiny toy things we've been working with.

Let me move on to the market issues. Basically the market issues focus on realizing our ultimate goal of moving hypermedia out of our labs, out of the research and development labs, and into wide-spread use. Basically, as Bob Glushko's panel on the World's Collide, it was framed as what are the economics of hypertext and I think that's an issue that we have to solve. What are the economics of our field? And I've divided that into three issues: the problem of defining hypermedia markets, standards, and publishing hypertexts.

We'll begin with the problem of defining hypermedia markets. I always ask myself this question, if hypermedia is so great why isn't it selling like hotcakes? There are hypermedia companies and many of them are at least feeding themselves but where's our Lotus and where's our Lotus 1-2-3. Where are the big companies that are going to take this idea and really make something of it.

I think one of our problems is that the field is way too technology driven, that we just really don't understand our market and where this technology will be useful. I really like the Malcom et al. paper from Boeing on requirements but I was very surprised that it should appear in a research conference this late in the game. We should all know that stuff in that paper already and, therefore, we shouldn't have had to have that paper but, in fact, we did have to have that paper and the program committee was wise to choose it.

There's lots that we can choose from, a lot of different application areas we can choose from. Each one has a different requirement. Technical manuals are going to look very different from interactive fiction. But whatever market we choose we need to develop an understanding about the costs and benefits of using hypermedia in these applications.

I'm basically here just making a plea. I don't understand markets or marketing very well but we need to understand it better somehow. And I think that the issue of hypermedia marketing is a very critical one even for the researchers among us, even those of you who, like myself, like to spend our time in ivory towers have to pay attention to the hypermedia market. First of all, a user-centered technology like hypermedia needs users to drive the research. We can't just build systems and then say `well, that's nice`. We've got to get people using them to get feedback.

And second of all, with a reputedly useful technology like hypermedia, I truly believe as the market goes so the research funding goes. That if we don't build a market for our system, nobody is going to be interested in funding research on those systems because it's looks like a losing cause. So markets are important, even for researchers in the field.

Let me turn to Issue 2. I believe we need to start developing and using standards now. It's not too early to talk about certain hypermedia standards. Lots of places we can put standards in. Last keynote address in '89, Norm Meyrowitz challenged us to have anchoring protocols in our desktops. He said they'd be there two years from now, in fact, they aren't here. So it's time to standard on anchoring. Hyperdocument interchange - we've heard some very nice work at this conference from the HyTime people. I expect to see other ones, Link Services, addressing problems. The World-Wide Web, for example, which was in the Demos did a nice

example of inter-network addressing on hypermedia documents. So there's a need for all kinds of standards in the field.

I'd like to make a plea for standards based on well-articulated models, not necessarily formal models, but the models have to be well-articulated. You can't just promulgate a standard. You have to isolate out what the model behind that standard is, because I believe our field will change rapidly and the standards need to change along with it and if the standard is all details and no abstraction you won't be able to modify your standard appropriately. I also believe that our standards have to complement other standards from other fields and the prototypical example is the object management standards I spoke about under the open systems issue. We can't be developing standards separate from standards in a whole lot of other different fields. But, however we approach the problem we need to start working on standards now - and more than just HyTime standards.

Let me turn to the final issue, market issue, is publishing hypertexts as opposed to hypertexts publishing which is what Ted Nelson would like to talk about. I think we have a classic bootstrapping problem in our field. There is little demand for published hypertexts because there are so few published hypertexts around. On the other hand, there are so few published hypertexts around because there is little demand. So the question is how do we get this vicious cycle to end? The question I always ask myself is 'where are the Eastgate Systems of the world?' Basically, if you've been at this conference you know what Eastgate Systems is. It's Mark Bernstein's company that does a very nice job of publishing hypertexts. You know they sell a number of these interactive fictions and a whole lot of other things. And I'm wondering why we aren't seeing more such companies publishing hypertext in the world. And I think we need to do that.

I think increased publishing will require more standardization. And where that standardization will come is an interesting issue. Do we want to standardize on a delivery system, should everything be sent out in Storyspace, for example. I doubt we'll ever get agreement on that. Should we standardize on an interchange format, for example, HyTime. So it's possible that might be a way to go. I think much more interesting is to look at standards for material preparation. The text encoding initiative in the humanities and the Perseus project - have looked at this notion that you want to prepare your materials in a standard format and then it can be compiled into various delivery or interchange formats. But it's the materials preparation standardization that needs to be done.

Publishing hypertext will raise a whole lot of interesting issues. We heard at this conference from Samuelson and Glushko about some of the copyright issues in publishing hypertext and I expect there's going to be a lot more activity in that. We heard in '87 a very interesting talk which I don't think really got published by Greg Crane on how a classics professor basically can't get tenure by publishing hypertexts. But I think it's probably true of a computer science professor for that matter these days. But I think until we solve the tenure problem, I mean who produces most of the paper in the world? It's probably academics and we have to solve this problem. There will be a whole lot of issues, the economics of publishing of hypertexts, and so on. But we need to start seeing more hypertexts in the world to bootstrap our field.

So basically I think we have a dual agenda for our field in the next few years. For research and development in our field, we have to solve some of these technical issues and there are a lot more. I had a list originally of 17 and I cut it back to seven technical issues. And we also need to pay attention to these marketing issues or these issues of moving out in the field.

Retraction and Conclusion

Let me just end up with a retraction. I do not believe anymore that hypermedia is headed toward obsolescence whether it's self-obsolescence or otherwise. I hope my talk has convinced you, and I'm certain this conference has convinced you, that the field is alive with enough challenges and exciting ideas to keep it interesting and moving for many years to come and I certainly look forward to seeing the progression on these ideas and some of my new issues and the introduction of many new ideas next year at ECHT '92. And if I'd known I would have put a dot right there in Milano. Thank you very much.

Mark Bernstein bernstein@eastgate.com (617) 924-9044
Eastgate Systems, Inc. <http://www.eastgate.com> (800) 562-1638
134 Main St.
Watertown MA 02472 USA <http://markBernstein.org>

@eastgate