

**Footprints:
Interaction History for Digital Objects**

by
Alan Daniel Wexelblat

B. S. E Computer Science, B.A. Philosophy and Science
University of Pennsylvania
Philadelphia, PA, 1984

M. S. Media Arts and Sciences
Massachusetts Institute of Technology
Cambridge, MA, 1994

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in Partial Fulfillment of the requirements of the degree of

DOCTOR OF PHILOSOPHY
at the
Massachusetts Institute of Technology
February, 1999

© Massachusetts Institute of Technology, 1999
All Rights Reserved

Signature of Author

Program in Media Arts and Sciences
January, 1999

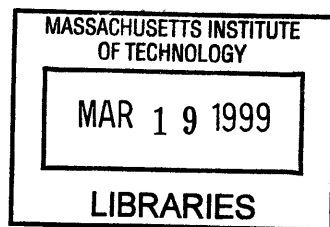
Certified By

Professor. Pattie Maes
Associate Professor of Media Arts and Sciences

Accepted By

Stephen A. Benton
Chairperson

Departmental Committee on Graduate Students
Program in Media Arts and Sciences



ROTCH

Footprints:
Interaction History for Digital Objects

by
Alan Daniel Wexelblat

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning in January 1999,
in Partial Fulfillment of the requirements of the degree of
DOCTOR OF PHILOSOPHY in Media Arts and Sciences

Abstract: Digital information has no history. When we interact with physical objects, we are able to read the traces left by past interactions with the object. These traces, sometimes called “wear,” form a basis for the interaction history of the object. In the physical world, we make use of interaction history to help come up with solutions and guidance. This is not possible in the digital realm, because the traces are missing. This dissertation describes a theoretical framework for talking about interaction history. This framework is related to work in anthropology, ethnomethodology, architecture, and urban planning. The framework describes a space of possible history-rich digital systems and gives properties which can be used to analyze existing systems. The space consists of six properties: proxemic/distemic, active/passive, rate/form of change, degree of permeation, personal/social, and kind of information. We also present an implementation of these ideas in a system called Footprints, a toolset for aiding information foraging on the World Wide Web. Our tools assume that users know what they want but that they need help finding it and help understanding — putting in context — what they have found. Footprints is a social navigation system, designed to show that information from past users can help direct present problem-solvers. We present results from informal use of the tools over the last two years, and from formal surveys and experiments on a controlled task. These experiments showed that people could achieve the same or better results with significantly less effort by using our tools.

Thesis Supervisor: Prof. Pattie Maes
Title: Professor of Media Arts and Sciences

This work was supported in part by IBM, Telecom Italia, AT&T, and the Media Lab’s News in the Future and Digital Life consortia.

The views and conclusions contained in this document are those of the author and should not be interpreted as representing the policies or views, either expressed or implied, of the sponsoring agencies or any other person or organization mentioned within.

**Footprints:
Interaction History for Digital Objects**

by
Alan Daniel Wexelblat

The following people served as readers for this thesis:

Reader

Walter Bender
Senior Research Scientist
MIT Media Laboratory

Reader

Dr. Stuart K. Card
Research Fellow and Manager,
User Interface Research Group
Xerox Palo Alto Research Center

Reader

Dr. Jim Hollan
Professor of Cognitive Science
University of California, San Diego

This dissertation is dedicated

to all my teachers

both formal and informal

*Whoever would walk the Path must become the Path itself.
the Buddha*

Acknowledgements

Many dissertations are the story of one person's struggle against overwhelming odds to produce the end result. This is not one of those. Rather, this is a story of someone who has been blessed with extraordinary amounts of help, love and support. I would like to thank all of those who have contributed so greatly towards making this work possible. Unfortunately I will probably miss many of those who have helped, simply because there have been so many.

I must begin with my parents, and my family. My parents have directly and indirectly supported first my decision to initially put off getting my Ph.D. then my decision to return to school. Their love and support has been invaluable. My mother has been kind enough to try and help me untangle my overly technical and confusing prose. Her editing has improved this document no end; any remaining errors and omissions are my fault, of course.

Next, my thanks goes to Dr. Pattie Maes, my boss and advisor for the last four and a half years. She has been incredibly patient with me as I worked out the problem on which I would eventually focus. She took a risk on me, based on not much more than a few conversations one summer long ago. Earning her ongoing respect has been very important to me and I am proud to be the second Ph.D. student graduated from her group.

I have also been blessed with wonderful committees. My generals committee — Dr. Ken Haase and Dr. Whitman Richards — helped me clarify my thinking and kept me from making any number of egregious errors. My dissertation committee — Walter Bender, Dr. Stuart Card, and Dr. Jim Hollan — has provided invaluable feedback and played a key role in shaping this document. I have also benefitted from the advice and experience of an unofficial committee: Dr. Henry Lieberman, Dr. Tom Erickson, and Dr. Ted Selker.

Those of you who are at or have been at MIT know that nothing gets done without the help of the undergraduates (or UROPs, as they are called, for Undergraduate Research Opportunity). In my case I have been blessed with a cadre of truly wonderful UROPs in my years here. Particular thanks goes to Felix Klock, Alex Lian, and Jim Matysczak who have been with me ever since Version 1 crawled from the primordial ooze of my imagination into reality and without whose incredible coding efforts none of the later versions would ever have seen the light of day. Zahra Kanji helped run the user tests, and Usman Mobin wrote the program to analyze the test log data.

A special thanks is reserved for Linda Peterson, goddess of the Media Lab. Without her intervention and assistance I would never have gotten in, nor been able to remain here. I also want to thank the other faculty and students of the Lab, particularly the students in the Software Agents group for their feedback, friendship and support, and Dr. Roz Picard, geek, asb, Warren Sack, Marc Davis, Dr. Henry Jenkins and the members of NI.

A great boon of understanding has been bestowed on me by my housemates and my officemates, all of whom have been willing to tolerate my odd hours, loud laughter and louder cursing. The format for this dissertation was lifted from prior publications by Kevin Lynch and Edward Tufte. I, like all the people who work here, have benefitted from the assistance of the people who build, maintain, and keep in shape the computer systems, both the formal work of NecSys and the informal work of the agents-sys people.

Finally, there are no words to express my debt to Michelle. Her companionship, love, and support have not only made it possible, they've made it worthwhile.

This document was prepared in Frame Maker 4.0 on Unix and 5.0 on PC platforms. Times Roman typeface is used for body text. Pictures were scanned and edited in Adobe Photoshop and the results recorded as GIF files. Annotations to the pictures and additional diagrams were done with Frame Maker's internal drawing tools.

CHAPTER 1 *Introduction* **15**

Collective Intelligence 17
Contributions of this Dissertation 18
Roadmap to the Dissertation 18

CHAPTER 2 *Theory* **19**

Motivation 19
Interaction History 19
Interaction History Framework 21
Using interaction history 32
Skilled practice and interaction history 33
Satisficing and interaction history 34
The first reflective conversation 36
Interaction History as the Second Conversation 38
Social Memory 40
Organizational Memory 42
Summary 44

CHAPTER 3 *Navigation and Navigation Tools* **45**

Motivation 45
Alternative Model of Navigation 46
Lynch's Elements of Navigation 47
Relationship of Navigation & Design 49
Social Navigation 51
Footprints Tools 52
Footprints on the Web 53
Enhancing the Data Structure 56
Maps 68
Paths 74
Annotations 80
Comments 81
Purposes 84
Summary 88

CHAPTER 4 *Related Work* **89**

Related interaction history work 89
Social Navigation Systems 95
Structured Navigation Systems 100
Non-central authorities 102
Architectures 104
Summary 105

CHAPTER 5 *Results* **107**

Insights 107

Survey 108

Experiment 111

Summary 123

CHAPTER 6 *Conclusions and Future Work* **125**

Improvements 126

Conclusion 131

FIGURE 1.	Two Forms of Accumulation (from Krier)	20
FIGURE 2.	Accretion and fade functions	24
FIGURE 3.	Using plans in real conversation	37
FIGURE 4.	Park Street Station	39
FIGURE 5.	Wear in the station (floor closeup)	40
FIGURE 6.	Framework for analyzing remembering in work (from Engeström)	43
FIGURE 7.	Top-down (left) and Bottom-up (right) hierarchies	51
FIGURE 8.	Stylized representation of HTML “frames” page	56
FIGURE 9.	High-level Footprints Implementation Architecture	57
FIGURE 10.	Footprints Operation Sequence	58
FIGURE 11.	Footprints Control Panel	59
FIGURE 12.	Access log line	60
FIGURE 13.	Instrument log line	60
FIGURE 14.	Detailed Architecture of Footprints front end	61
FIGURE 15.	Etrans	63
FIGURE 16.	Cluster	63
FIGURE 17.	Typical False Start Pattern	64
FIGURE 18.	Distribution of Dwell Times	66
FIGURE 19.	Possible Methods of Combining Two Sequences	66
FIGURE 20.	Footprints Standard Site Map	67
FIGURE 21.	Footprints Extended Site Map	68
FIGURE 22.	Footprints Complete Map	69
FIGURE 23.	Cluttered Map	70
FIGURE 24.	Footprints Map Titles Window	71
FIGURE 25.	Dimensions of Navigation Experiences	72
FIGURE 26.	Narrative Experience Example	72
FIGURE 27.	Storefront Experience Example	73
FIGURE 28.	Message Experience Example	73
FIGURE 29.	In-and-Out Experience Example	74
FIGURE 30.	Footprints Paths Viewer	76
FIGURE 31.	Multi-highlighting in the Path Viewer	77
FIGURE 32.	Three Views of a Web-based Publication	78
FIGURE 33.	Who is Pattie Maes?	79
FIGURE 34.	Footprints Annotated Web Page	81
FIGURE 35.	Comments in the Path Viewer	82
FIGURE 36.	Comment Window	82
FIGURE 37.	Comment-entry Window	83
FIGURE 38.	Purposes Tool	84
FIGURE 39.	Paths Page	85
FIGURE 40.	Footprints Tools in the Interaction History Design Space	87
FIGURE 41.	Cross-condition Comparison of Satisfaction	115
FIGURE 42.	Typical Experiment Map Cluster	119
FIGURE 43.	Fruits of Infoseek	120
FIGURE 44.	Edmund’s Map Cluster	121
FIGURE 45.	Edmund’s Paths	121
FIGURE 46.	The Boston-local Effect	122

The dragon path ran straight down the alley. Vesica could feel it now, or imagined she could. An electric chill, swirling around her ankles. And more than that: She could feel other patterns of energy laid across the path, or trapped there. Impressions left like fingerprints by people who had passed this way. Scraps of time the path had snared the way brambles tear off pieces of your clothing. Moments imprisoned in amber.
Richard Grant

Reference 29

Interaction history, the record of how objects have been used in the past, can be used to help organize behavior, solve problems, and ease tasks in the present. To do that, we need first to understand how interaction history is used in the physical world, then consider how we might apply it in the digital realm.

This dissertation is, in many ways, about stories — everyday stories we construct to tell each other and ourselves how to deal with and how to make sense of the world full of artifacts around us. It is not about stories that emerge by preplanning — as when a teller organizes, top down, a tale for instruction, edification or amusement — but about stories that emerge from the bottom up, told by people collaborating with each other on the construction. It is about stories situated in the world and their role in connecting us to the objects of the world and to the people who have interacted with those objects before us. We call this “interaction history.” The dissertation begins with a theory of what interaction history is and why it is important to expert problem-solving in the real world, and includes a description of a toolset (Footprints) built to explore the application of interaction history to problems of finding and comprehending information in a complex environment.

Every real object that has been put to human use tells stories. We learn how to read these stories as part of our cultural and social upbringing or as part of professional training. We become so skilled in doing this we forget that it is happening all the time all around us, a fact that is essential to our survival as culturally capable humans. Throughout this dissertation I will try to call the reader’s attention to instances of these kinds of stories — in effect, pointing out the obvious.

To begin with, imagine that you decide to learn something about the programming language Java. You’ve heard the name, but now you want to know more about the details. You might do this by going to a bookstore and buying one of the multitude of available books on Java. Or you might go to a facility where Java is used, such as the MIT Media Laboratory, and borrow a copy of the same book on Java. Now you have two objects: in one hand is the bought book and in the other, the borrowed book. From a classical information retrieval point of view the two objects are the same — they have the same words, the same pictures and diagrams, and so on. In reality, though, these two books are very different. The borrowed book usually comes with

The “Book on Java” Story

something more than just the printed text. It may have underlined passages, dog-eared pages, notes in the margin, highlightings, and so on. Even the physical object is different — the borrowed book probably opens easily to pages where it has been propped open. The borrowed book comes with a history; it is a “history-rich” object.

Although the history may degrade the object to some extent — for example, a careless underline may actually obscure text — it is this interaction history that tells the book’s story, accumulated through a multitude of interactions between people and the object. The real, physical world is full of history-rich objects. Both such objects and the ways we interact with them have evolved over time. The effect of this evolution has been predominantly to smooth out our experience of history. Interaction history generally does not obtrude; it is normally subtle and unintrusive. The ways in which interaction history gets recorded, filtered, redisplayed and used have become so transparent that we think of them as “natural” even though they must have been very strange at one point in our society and they must be experienced as strange by children during their cultural growth. In fact, they are the end-products of lots of trial, error, and long-forgotten awkwardness.

*Digital information has
no history*

Unfortunately, digital information (by and large) has no history. Where physical objects provide natural media for recording interactions, we need additional tools to record history of interactions with digital data. Unfortunately, these tools have not been created. Digital data tell no stories. When we open a spreadsheet, a word processing document, or a Web page, we often have the sense that we are the first and only person ever to use these data. What happened in the past? Who used it? How was it used? Why were actions taken, if they were or, if not, why not? None of these facts is available, and usually few or none of them can be reconstructed.

Interaction history is more than simply the changes in state of an object. State changes may or may not be meaningful, and some important interaction history leaves no trace on the object but depends instead on our relating objects to other objects, to external sources of knowledge, or to temporal and use contexts that are not part of the formal record. Often we want to know not just what was done, but by whom, when, in what manner, and for what reasons. Chapter 2 of this dissertation develops a framework for interaction history, based on such purposes.

*Interaction history is
social*

Interaction history is also a fundamentally social process. Even if an object has always belonged to, and only ever been used by, one person it has been used by that person over a period of time. This extended accumulation allows the past selves to “talk” to the present self, just as they would to another person. This social process encounters all the usual problems of communication at a different time: ambiguity, incompleteness, loss of sense-making context. How often have we wondered at our own cryptic margin or class notes, just as we wonder at the apocrypha others have written? Collective, communal or social knowledge also benefits from the variety of the participants’ backgrounds, their joint knowledge, and their experiences.

Sometimes this variety can lead to conflict, as participants in the interaction history who come from different backgrounds see things differently. Take for example, the much-publicized trial of O.J. Simpson. One person reading a story on the trial might see it as yet another example of a husband brutalizing his wife and getting away with it. A different person might see it as a story of how a rich man is able to “buy” justice. Still a third person might see it as a story of the white establishment’s attempt to take down the most well-known black person in the city. No interpretation is any more wrong than it is right — conflict emerges when tools, the situation, or the environment try to force one meaning on the situation.

Interaction history, if done correctly, is the antithesis of such forcing. History that works takes into account what has been done before and allows for multiple possibly conflicting futures to emerge. The assumption of interaction history is that there is no one right or perfect solution to a problem. Instead, the assumptions are that there are multiple satisfactory solutions, that diversity of answer possibilities is a benefit, and that good comes from a useful coverage of the possibilities.

Together we know everything

The underlying assumption has been called “collective intelligence,” an idea that is summarized by the slogan at the head of this section. This slogan is used by the PHOAKS (People Helping One Another Know Stuff) project. Fundamentally, if there is no authoritative source for an answer, it may be that the answers are to be found distributed in the community knowledge. This, however, begs the question of how we transmit such knowledge. In a small enough community, where the members are in close and regular contact, it may be simplest just to go around and ask people until one finds an answer. In other cases, answers may be found by asking any of a number of people, possibly just by throwing the question out into the “ether” and knowing that it will be seen by enough knowledgeable people to have a good chance of being answered.

Most of the time, however, there are too many people involved, and they are too distributed over space and time to make these methods possible. Instead, we must have other means for transmitting questions and answers. Interaction history is one possible mechanism; the community records its answers in the process of each person solving his or her own problem. The work done by any one person is unlikely to be particularly useful to any one other specific person. But the collected result is likely to provide a satisfactory solution for most users.

A simple example of this from the physical world may make the notion clearer. There is a story, possibly apocryphal, that the campus at the University of Michigan was designed with only buildings and no walkways. The buildings of the campus were used for a year and then in the summer, the contractors came back and laid down sidewalks between the buildings. However, instead of laying walkways according to some top-down master plan, the story is that the concrete was poured in those places where people had walked. Their steps had beaten down the grass between the buildings in just the ways that people traveled among the offices and classrooms. The campus, without requiring additional conscious effort from the people, had recorded the interaction history of their travels. This history was then used to make permanent paths. Asking any one member of the Michigan community what were the optimal paths among the buildings would have yielded only a small fraction of the paths that were embodied in concrete. And that small set of paths would have been unlikely to be useful to any other particular user. The collective solution, though, is likely to satisfy almost any user, because it collects the work of thousands of people.

Interaction history, then, is not a radically new idea in the world. Rather, it is an attempt to capture techniques and ideas pervasive in the physical world and take advantage of these techniques in the digital world. As with any such endeavor, it is important to define the scope of interest, characterize it, and describe where and when such techniques would be useful. This dissertation attempts to do these things, and to relate my application of interaction history techniques to one particularly apt problem — navigation in a large complex information space.

1.1 Collective Intelligence

Reference 77

See “PHOAKS” on page 99.

Reference 1

Walkways at Michigan

1.2 Contributions of this Dissertation

This dissertation describes a theoretical framework for talking about interaction history. This framework is related to work in anthropology, ethnomethodology, architecture, and urban planning. The framework describes a space of possible history-rich digital systems and gives properties which can be used to analyze existing systems and to think about problems and tradeoffs which must be dealt with for future history-rich systems. The space consists of six properties: proxemic/distemic, active/passive, rate/form of change, degree of permeation, personal/social, and kind of information. We also present an implementation of these ideas in a system called Footprints, a toolset for aiding information foraging on the World Wide Web. The toolset includes three methods for displaying history information. Maps give high-level overviews. Paths give lower-level views relevant to the user's current location. Annotations show history directly on the Web page. In addition to the display tools, there are tools to allow users to make comments and to express their reasons for going to Web pages.

Our tools assume that users know what they want but that they need help finding it and help understanding — putting in context — what they have found. Footprints is a social navigation system, designed to show that information from past users can help direct present problem-solvers to gain more efficient and more satisfying answers to their problems. We present results from informal use of the tools over the last two years, from a formal survey and from an experiment on a controlled task. These experiments showed that people could achieve the same or better results with significantly less effort by using our tools. We also found that interaction history can be used by information designers to improve their organization of information, and by information users to improve their comprehension and navigation.

1.3 Roadmap to the Dissertation

As you, the reader, have no doubt noticed by now, this dissertation is formatted somewhat unusually. My goal is to have the format itself serve as an example of interaction history. You will have the chance to navigate this document in part by seeing the notes I have made, and that have been suggested by my committee members and by other readers of this document. These notes, section titles, and references to literature materials, are separated out from the text for ease of spotting and for quick navigation.

Chapter 1, this chapter, serves as the introduction to the concept of interaction history. Chapter 2 is the theory chapter. It attempts to frame the theoretical portion of this work in terms of theories already existing, particularly with relationships to certain theories in cognitive science, ethnomethodology, and architecture/urban studies. It introduces the framework of 6 properties that is used to describe history-rich artifacts and systems. Chapter 3 discusses the problem domain we chose to focus upon — navigation and design for navigation — and describes Footprints, the software tools built to explore interaction history theory. Related work in navigation systems and other software implementations of history-rich systems are covered in Chapter 4. Evaluations of the tools, both formal and informal, are presented in Chapter 5 along with some of the insights gained and limitations learned. Finally, Chapter 6 gives analyses, conclusions, and directions for future work.

CHAPTER 2 *Theory*

[Theory] is not simply a framework, though it can provide that; rather, it is a story about what you think is happening and why. A useful theory is one that tells an enlightening story about some phenomenon.
Joseph Maxwell

Reference 48, pp32-33

To continue with our story theme for this dissertation, this chapter may be considered the story of the theory of interaction history, in the sense that Maxwell speaks above of theory stories. That is, the chapter attempts to form a coherent description of a new field of study — to define an organizing framework for that theory, to explore connections between this new theory and existing theory, and finally to lay the groundwork for the future chapters, which will describe some ways the theory may be applied.

The goal of this chapter is to develop a framework for talking about interaction history. The framework has two parts — a set of properties that give us a language for talking and thinking about interaction history and a set of related theories that give us an understanding of how interaction history is created and used in the world. The properties form a kind of multidimensional space, within which we can locate interaction history systems. Theories of personal and social action to be discussed will show the integral nature of interaction history to human environmental cognition and skillful operation in the real world. This framework provides us with both the retrospective ability to analyze existing systems (as we will do in Chapter 4) and a way to think prospectively about relevant design issues in building new history-rich systems.

2.1 Motivation

Our goal is to find ways to make interaction history information as easily and naturally available in the digital world as it is in the physical world. This effort is inspired by original work done by Hill and Hollan in the early 1990s. To do this we first want to establish a theoretical basis, then build systems to experiment with out theories. Later chapters describe our systems and our experiments with them.

Reference 34, 35

Interaction history is the accumulated record of the interactions of people with objects. It encompasses three categories of representations.

2.2 Interaction History

1. Sequences of actions, relationships of elements on which people have acted, and the resulting structures. Sequences are time-related actions which form a purposeful variation. For example, assembly instructions, or the directions in a cookbook recipe. Relationships of elements are structures that encode temporal information, such as ordered lists or directed graphs. Finally, resulting structures are modified, unique

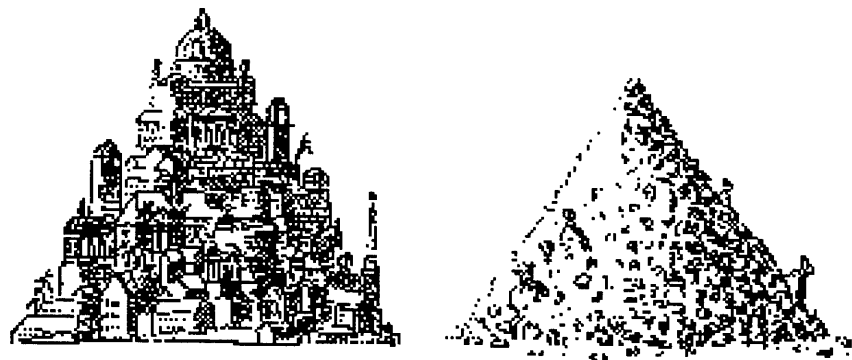
Action sequences

objects. Every house in a subdivision may be designed to be identical, but as soon as real people begin using real tools and materials to build the houses, the structures begin to vary from the plan, and from each other. In this form of representation, time is used as a presentation strategy; we depend on observers' perceptual or conscious memories to make the connection between the current item and the ones shown before or above it.

Reference 8
Reference 41, page 36

History modifies an object's state, producing a visible accretion, but not just any accumulation of state modifications are interesting history, as illustrated in Figure 1 below. This diagram is reproduced from Stuart Brand's *How Buildings Learn* and credited to Leon Krier. The picture shows a city organically emerging from and on a hill as opposed to an equal sized mound of undifferentiated "stuff." Each represents an accumulation, but only the left-hand accumulation has history (of the sort that interests us) because it represents the results of many purposeful interactions.

FIGURE 1. Two Forms of Accumulation (from Krier)



In the digital world, companies find themselves overwhelmed with stuff like the right-hand side of the picture: mountains of raw data, such as web-server click streams or customer purchase records. Most of these data are never examined by mechanical data-mining techniques, let alone by human eyes. It lacks the kind of relationship structure to make useful history; at most one can find isolated statistical correlations in it.

Temporal collage

2. Temporal collage. Multiple states are shown directly, as in "before and after" pictures used to show dramatic changes. The situations depicted in the before and after pictures existed separated by some amount of time, but they are shown simultaneously, as if they had existed together. In this form of representation, time becomes part of the content. By virtue of the juxtaposition — and the montage thereby created — we do not need to depend on users' temporal memories.

We can create tools or technologies to allow users to infer the process involved in objects undergoing modifications, that is, changing from one state to another. This is particularly useful when the change would otherwise be too fast or too slow to be perceived. Of course, users who are more versed in the field may understand the underlying process simply by virtue of being experts. Temporal collage is often useful when there is a desire to separate the "original" or created object from its current incarnation or when responsibility for sets of changes must be assigned. For example, a document produced by a writer might be shown simultaneously with several modified forms, each showing changes made by one of the editors. Finally, the complete edited document with all changes made and blended into the text could be shown.

3. Recurrent opposed states. History can be inferred from the contrast of the present state both with the remembered past state(s) and with the expected future state. For example, when I edit a source-code controlled file in Emacs, it displays the legend “CVS:1.6” as part of the edit mode line. This bit of history reminds me that the source file is “checked out” from the version control system, and raises the expectation that I will check it back in at some point. In the physical world we can look at a tree and see that its leaves are green, but know that soon they will turn colors, wither, and fall off. Likewise, the bare branches of a tree in winter hold the promise of leaves, because we know that these states recur.

Recurrent states

Recurrence implies a rhythm, which may be invertive or cyclical. An invertive display is something like a “blink comparator” such as is used by ophthalmologists when determining a person’s prescription. Two different possible adjustments are shown in rapid and repeated succession and the patient is asked which is better or clearer. Cyclical states imply a sampling interval and a presentation interval; these are harmonized to give the rhythm of the representation. We may sample trees in four views because this matches to our cultural notions of the year being divided into four seasons. A farmer may think in terms of planting, growing, harvesting and fallow seasons; a software developer may think in terms of design, development debugging, and release cycles.

The modifications or additions that make up interaction history affect not just the object but our perceptions of, and uses for, the object. For example, near the end of 1997 news stories began appearing indicating that works of art that were featured prominently in public American museum collections had been plundered by the Nazis. This bit of history did not change the objects themselves in any way. Instead it changed our perceptions of them. What had been done in the past changed our ideas on how to treat the objects in the present.

Reference 12

We use interaction history information daily in the physical world. Objects often carry their histories with them, as in the example of the dog-eared book from Chapter 1. Physical objects often change in response to use: brass shows a patina where human sweat and friction have been applied, stone stair treads show curvatures worn by thousands of shoes passing over them, etc. We make use of these traces to guide our actions, to suggest possibilities, and to perceive solutions. This is often done without conscious thought, as when we let a book fall open to a particular page because it has been opened to this page often in the past.

See “The “Book on Java” Story” on page 15.

To characterize the problem space, this section presents a framework that can be used to talk about interaction history and to describe the space of all possible interaction history systems. Six major elements, or properties, describe this area.

2.3 Interaction History Framework

Ideally each of these elements would be orthogonal and could be said to be dimensions. However, the characterization is more complex than this. Several of the elements of the framework relate to each other; one is taxonomic rather than linear, and there are definite interactions among the elements. Thus, this framework has been extremely valuable in helping formulate history-rich work so far and we offer it as the best descriptive system we have to date. An important part of this research is finding useful points in this multidimensional space for history to be shown in computer interfaces. As we describe the properties we will give examples of how each is important for digital media.

Property 1

Reference 32, 33

Proxemic versus Distemic

Urban planning and social anthropology use the words “proxemic” and “distemic” to describe the closeness (proximity) or distance relationship of people and spaces. The original description of proxemic space, by Edward Hall in 1973, introduced the idea that there are four levels of proximity in space to a person engaged in an observational or communicative act. Hall’s levels are *intimate*, *personal*, *social* and *public*. Intimate space is where touch happens; personal space is used for conversations. Social distances are set up by groups of people and usually are less than ten feet. Public spaces go beyond this.

Reference 30

The idea of distemic spaces originated with Barrie Greenbie, whose thesis was that as we travel outside our childhood or normal cultures we begin to replace default unconscious behaviors with abstract communication and “rationally directed activities.” That is, within our familiar sphere, we operate as if everyone is like ourselves and all understand the same rules of spatial behavior. Once outside that sphere, we begin to rely on codes of conduct which must be external, formalized, and known to all people. For Greenbie, social diversity is synonymous with distemic, since diverse cultures cannot mesh except by commonly agreed sets of formal rules. Similarly, a measure of how distemic a space is can be had from looking at how much of the behavior in the space is governed by formal rules. For example, Greenbie points out that the modern automobile highway is a highly distemic space:

Reference 30, p. 115

Perhaps the clearest example of what I mean by a distemic space is an automobile highway, where anyone with a driver’s license is free to travel more or less as desired, provided a few relatively simple rules are followed. Even a bank robber will follow the most basic of such rules, such as driving on the proper side, because it is impossible to use the space for long without doing so.

For interaction history purposes, we consider proximity to be a function of the actual distance and the cognitive distance between the person and the space. A proxemic space is one which is felt by users to be transparent, in that the signs and structures can be easily understood. People feel comfortable in, or part of, the space. Operating in proxemic spaces is felt to be comfortable, smooth and natural. Conversely, distemic spaces are uncomfortable or opaque to users. Signals go unseen, usually because the people in the space lack the required background or knowledge to translate or comprehend what they experience. Operating in distemic spaces feels awkward and hesitant. We feel “close” to our bedroom even when far away from it and experience a certain “distance” when we sleep in someone’s guest bedroom.

Interaction history systems may be more or less proxemic based on how well they relate to their users and how well they take advantage of users’ past experiences and knowledge. For example, the Emacs mode-line history noted above (CVS:1.6) would be completely distemic to anyone who did not know that cvs is a concurrent version-control system that uses numerical markers to distinguish successive versions of a source file. A proxemic system is something like Netscape Navigator’s bookmarks which users can arrange, group, subdivide and categorize as they please. Of course, this proximity derives from the singular nature of the person’s arrangement. A proxemic group space is something like a collaborative workbench, where the tools are arranged to be easy to reach for all the people who use the bench.

One of the biggest challenges for designers of interaction history systems is to consider how we can make such systems proxemic when operating in a distemic environ-

ment. By their very nature, computer systems are distemic in that they require us to learn and follow potentially very rigid rules; they are also usually very unforgiving of mistakes and failures to follow their rules. In order to be productive with computer systems, we must learn these rules at least to some extent. As Brian Eno so aptly put it, “Familiarity breeds content.” Proxemic systems will be more familiar to users and will enable them to produce more content or solve their problems.

Reference 22

Active versus Passive

Property 2

Most interaction history is passive; it is recorded and made available without conscious effort, usually as a by-product of everyday use of objects. Conversely, when we stop to carve a signpost to leave on the trail, we are creating an active history element. As with property 1, the active/passive distinction relates to the user’s mental state and relationship to the history-rich objects. The sign is active in our vocabulary — even though it does not change or move without human intervention — because in order to make the sign we must stop and think of people who may follow us and read the sign. Thus we change our mental state from a passive one of simply going about a task to an active one of thinking about how the results of this task can be interpreted later by others.

When we prop open a cookbook so that we can read the recipe while our hands are busy, we do not think of our selves as recording a list of our favorite recipes. But that is, passively, what we are doing. We could actively record the same information, but this takes additional thought and effort. People often take the time to record history because they feel it is important. For example, we may make many editing changes to a document (passive history) before committing a new version to a repository (active history) where it can be seen by other people.

One of the advantages of the physical world is that most objects provide some means for passive recording of interaction history, usually by changes in the object or in its appearance. In the best cases, these changes enhance the object without destroying its functionality. However, even in cases where the interaction history progressively destroys objects, the idea of “well worn” is still valued. The challenge for history-rich computer systems is to find ways to allow interaction history to be passively collected so that users are not constantly thrown out of the cognitive state necessary to getting their tasks done. Of course, digital systems also have the advantage of allowing history to be made into a “first class” object so that histories can be duplicated and shared in ways not possible in the physical world.

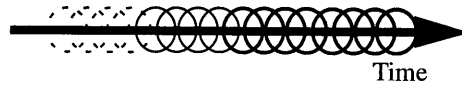
We can think of this as a “focus and context” property. That is, at one end of the spectrum, the user’s task is the sole focus and the history information provides the context. At the other end, users think specifically about the history information, and the context dominates (or becomes) the task itself.

Rate/Form of Change

Property 3

History moves forward, building as more interactions take place. I call this the “accretion” process since it is how history builds up. However, interaction history not only accretes, it also fades out. This can be roughly visualized by Figure 2.

FIGURE 2. Accretion and fade functions



One of the challenges for history-rich interfaces is deciding how to deal with this accretion. Just as a complete video playback of a meeting is usually not as useful as a summary, the total accumulation of history must be properly filtered, summarized or compressed so that it can be observed and used quickly. A good real-world example of this are ship navigation charts. These charts show summaries of thousands of depth readings taken by dozens of ships. These are summarized into a field of numbers oriented toward both expected courses of travel by ships in the charted area and toward highlighting potential danger areas, into which a ship might stray.

Reference 34

In the digital realm, Hill and Hollan's Edit Wear tool used a modified scrollbar to show areas within a source file which had been more or less heavily modified. Dozens or hundreds of accesses were summarized by an unobtrusive thickening of the "thumb" component of the scrollbar.

Fade function components

The fade function has three components, each of which can act independently: obscuring, losing and disconnecting. Taken together, the three components make up what we would call "forgetting." We separate them here because each of the components is amenable to different solutions.

Obscuring is the process by which new history blocks the perception of old. A new coat of paint obscures the wear on the previous layer. History is often layered and while this layering gives a richness to historical objects, it also covers up things which may be valuable.

Losing is the process by which interaction history is removed. Old objects may be replaced by new parts or, more seriously for organizational memory, people move on taking experience and knowledge with them. History-rich objects may also be outright discarded. This may be out of necessity, as when the object has worn out and can no longer be used for its intended purpose, or out of neglect, as when an object is found to have no present value and its historical value is not yet recognized. This is discussed in more detail below.

Reference 21

Disconnecting reflects the situation in which the interaction history is still available, but the connection between the history and its present interpretation has been lost. Engeström et al (1990) document this in Scandinavian hospital situations, where doctors' evaluation of patients' conditions does not match up with the notes written in patients' charts. The evidence (history) is still present, but its connection to the present has been lost. That is, the doctor is confronted with a patient who demonstrates or talks about certain symptoms; accompanying the patient is a chart which is supposed to account for the patient's current condition in terms of past reported symptoms, past treatments, current statistics, and so on. In theory it should be a simple process to look at the chart and understand how the patient comes to have the present set of symptoms; but in fact, Engeström found that doctors were unable to do this. They complained that the charts "did not make sense" in that they were logical but did not correlate with the doctors' understanding of the current situation.

Fading can be thought of as a decay function, much as we talk about decay functions for signals in which the strength of the history trace lessens over time. In interaction

history, there are two elements that make up the “signal strength.” The first element is the durability of the object. A diamond will be mined once, cut once or twice, and worn by a few people over many years. Conversely, a drinking glass will be cut once, used for a few years by a number of people, perhaps chipped or cracked, then discarded.

The second element is the durability of the trace. A fingerprint will not last long — usually only until the next cleaning or the next smudge overlays it. A carving will likely last much longer. The two strengths may not be additive, however. The fingerprint is no more likely to last on the diamond than it is on the glass. In some cases they may add: a carving in stone will likely last longer than one in wood. The durability of a trace is usually directly proportional to the degree to which it changes the object, or our perceptions of it. Our perception of the art stolen by Nazis is forever changed, presuming that our records and memories will outlast the paintings.

The trace

Both the accretion and fade functions will vary in any history-rich interface. A key benefit of modeling interaction history explicitly in digital systems will be to help users distinguish obscuring from loss, and to help prevent disconnection.

Degree of Permeation

Property 4

Permeation is the degree to which interaction history is a part of the history-rich objects. History may be inseparable from the object, as in the worn-stairs example, or it may be completely separate, as in the stolen-art example. In a history-rich interface, we must decide how closely to link the objects of interaction and the history information. In the physical world, we rarely have these choices, as history imprints itself directly on the objects of interaction. We can not use a book without opening it, though we may choose to use a bookmark instead of folding down the corners of pages. Digital data will only retain that history information we choose to keep; therefore, any record of this information must be captured and displayed by tools we create explicitly for that purpose, or by display systems built into existing tools; for example, the mode-line modification to Emacs described above. The particular set of tools we have built to display interaction history information (Footprints) is described in the next chapter.

This leads us to think about one of the greatest challenges of this thesis. In the physical world, interaction history highly permeates history-rich objects, and yet is peripheral to most uses of the object. As we consider integrating digital interaction history with the tools available to computer users, we would like to achieve a similar close integration between the data and the history; however, by making history closely permeate digital data, we risk destroying the very thing we are trying to enhance, because digital affordances are much less related to peoples’ physical skills than to the extent that they are embedded in non-physical representations.

In most cases, the history information must be kept peripheral, as it should be an enhancement to the user’s task environment. If interacting with the history information becomes a central focus, we have probably changed the task too much. On the other end of the spectrum, though, we cannot remove the history information too far from the user’s environment or one will not be able to take advantage of the accumulated work and expertise represented by the interaction history. Data which are removed must be displayed separately, and separate data risk becoming just another distraction on an already-cluttered display. Additionally, if we separate history from its object, we risk promoting disconnection, as described above. Conversely, separate history information can be used to provide what Peter Pirolli and his colleagues at

Reference 56

Xerox PARC have called information “scent.” The idea of scent is that from a remote location you can pick up an idea or trace of what your goal might consist of. Obviously scent can be used to give remote cues about what a more permeated history would provide.

Carrying Capacity

Reference 78

When thinking about designing representations for interaction history data, one of the most important questions is what we call the “carrying capacity” of the representation and of — what Edward Tufte calls the data density. In the physical world, the constraints of permeation limit what can be shown; for example, a hammer cannot be dog-eared in the way that the pages of a book can. In the digital realm, designers must be aware that the representation is relatively impoverished. Most displays do not support the density of representation available in the physical world. Even such simple things as anti-aliased fonts are often unavailable. We may try to use cues such as size, color, or line thickness to replace physical cues. Where they are available, though, we must keep in mind that these cues may already be used by the information designer. To the degree that we reuse them, we risk obscuring or destroying what we are trying to enhance. Thus, we say the carrying capacity of digital objects is potentially much lower than that of physical objects.

Likewise, we have to consider the carrying capacity of the separate history representations we create. We can make displays that are potentially very complex but we risk creating systems which will only increase the difficulty users have in connecting the history information with the source data. Overall, the carrying capacity is directly proportional to the amount of permeation possible: objects with no carrying capacity cannot have history permeated in them and must be augmented with separate representations.

An appropriate analogy for what we are trying to accomplish here is to a painting: the painting may be appreciated for what it is alone, or it may be framed. A good frame will enhance our interaction with the picture; it complements the art. Interaction history is like the frame; the picture is the information to which the history relates.

Property 5

Personal versus Social

History can be intimate to a person — what have I done? Or it can be social — what have people done here? Many tools focus on personal histories; for example, the facilities such as bookmarks within web browsers that allow users to see and revisit sites they have been to in the recent past. Group histories are more rare but, we believe, more valuable because most problem-solving tasks are collaborative in nature. One of the primary benefits of interaction history is to give newcomers the benefit of work done in the past. In fact, the slogan for our project is:

We all benefit from experience, preferably someone else's.

The other value of social history is that it promotes organizational learning. While it is true that organizations cannot learn unless individuals within the organization learn, it is also true that if the learning is confined to individuals then it is unlikely to benefit the group as a whole.

Reference 59

As described above, we consider memory and learning to be inherently social. Alan Radley notes that, in particular, the artifacts with which we interact on a regular basis create forms of social practices. That is, we build and continue to use precisely the kinds of objects that enhance the social rituals essential to our maintenance of home and workplaces. It is not coincidence that the writing desk and the social practice of writing personal letters disappeared at about the same time. The world of material

artifacts governs our relationships with the past in socially significant ways. For example, a woman may choose to wear a particular out-of-current-fashion wedding dress because it is the dress which was worn by her mother or grandmother at a past wedding. The history of this artifact changes the social relationship not only for the wearer, but for the audience at the wedding, which gains understanding and insight into the bride's relationship to her family.

Peoples' ways of remembering may also be different, depending on their relationship to the community, including the world of objects that are produced and preserved by that community. Julian Orr points out how the large number of minor variations and rapid technological changes in the photocopiers led to the rise of an oral story-telling culture among the copier repairmen. History-rich objects must be understood not solely as mnemonics — in which case any one object could substitute for any similar object — nor simply as triggers for memory, in which case a reminder of the object would be as effective as the object. Instead, history-rich objects must be seen as the basis from which we remember. The object is an active participant in the social construction of the past.

Reference 54

This is not to say that personal interaction histories are insignificant. On the contrary, an ability to see and review a record of what we have personally done in the past (and perhaps why we did it) can be invaluable. Anyone who has had to go back and review or enhance his own source code after months or years of not working on it can attest to this fact!

Kind of Information

Property 6

There is an infinite variety of kinds of interaction history information that can be captured. What kinds of information are important, to a large degree, are dependent on the task the observer is trying to accomplish. For example, a doorknob may show a particular pattern of wear accumulated over time, which may be important to someone interested in how people have handled the knob. Alternatively, someone interested in the history of the building might take note of whether the knob was made of expensive brass or cheap aluminum. In the digital realm, most code repositories only keep track of what can be searched for by relatively precise criteria. If I am looking for a particular algorithm, this information is useful. However, if I am trying to decide which of several implementations of similar algorithms is better, I need very different information — for example, ratings of the code by past users.

Don Norman has promoted the importance of the study of affordances within the human-computer interaction community. Affordances are the ways in which objects present themselves to humans for use. A handle, for example, presents a very different affordance for opening a door than does a button. The kinds of affordances available with objects controls how people will use them; screwdrivers afford use as hammers, often to the dismay of screwdriver manufacturers. Interaction history in digital systems will be the result of people manipulating whatever affordances are created.

Reference 52

Since we cannot possibly characterize all the kinds of information available, we focus on the uses to which interaction history might be put. We categorize the kind of available information loosely into *what*, *who*, *why*, and *how*. We then assert task/problem categories for which that kind of interaction history information would be valuable. Knowing what was done can be useful if users are searching for value, particularly among clutter. The underlying assumption is that users will converge on solutions

See “Satisficing and interaction history” on page 34.

that satisfice¹. Interaction history provides precisely the kind of rough guidelines that are valuable in satisficing. This is discussed in detail below. Additionally, the kind of information included in history-rich systems interacts with our choice of which of the three basic representation types we might use. For example, in trying to explain why something was done, it may be important to emphasize certain steps or events; this can be hard to do with a recurrent display. Alternatively, knowing how something was done may be very easily shown by a recurring display that touches repeatedly in the important elements in the sequence.

Knowing ‘what’

Knowing what was done can provide reassurance. If I want to be reassured about the reasonableness of what I am doing, records showing that others have done it before can reassure me that I am not alone. This is particularly helpful for novices who lack the kind of “reasonableness” guide lines that come with practice.

Knowing what was done can also give guidance; that is, the process of directing someone in a task or journey. Written directions can be guidance, and a record of what people have done before is also a form of guidance.

Knowing what was done provides structure and allows us to make sense of a mass of possibilities, thereby reducing clutter. History can perform an organizing function. I will argue in later chapters that history provides an externalization of peoples’ mental models and while any given user’s model may not precisely match with any others, it is still an organizing framework to which one can react or relate.

Knowing ‘who’

Knowing who has done something, or who is involved in the interaction history, can be important as well. Companionability refers to our natural desire to do things with our friends (companions). If I know what my friends have done I may be more likely to do it; conversely, I may avoid doing what my enemies have done, unless I want to know how they see the world.

This is related to sociability, our human tendency to do things in groups and to identify with groups even if no specific members of the group are present or known. For example, a Jew such as myself might have a greater interest in doing things knowing that they have been done by other Jews in the past, even without knowing the specific Jews involved. Similarly, a musician might want to retrace historical musical history, a chef might want to recreate historical recipes or cooking techniques, and so on.

This is similar to re-creation, the desire to do again what someone famous has done in the past. People in the Boston area often follow the path of Paul Revere’s Ride because they are seeking to re-create a famous historical event. Americans might similarly seek to sleep in a bed once slept in by George Washington.

Knowing who has done something is also important for establishing authority and possibly authenticity. A margin note written in the author’s own handwriting may be considered more relevant or authoritative than one written by an unknown reader. In some cases, authenticity and authority may be the same thing; for example, a baseball can have a scrawl on it which looks like Babe Ruth’s signature, but unless the scrawl was actually produced by Ruth it is probably worthless.

Knowing ‘why’

Knowing why something was done can be important for three reasons. The first and most common one is similarity of purpose. That is, I may care a great deal about

1. Are good enough given constraints of time and resources.

something that was done by people with a goal similar to mine. Knowing that the pages in a reference book were dog-eared by someone writing a paper similar to my current writing would be potentially important. Purpose here refers to human intention and so similarity is necessarily a loose measure based on the situation. My neighbor and I may both be searching for a mechanic to fix our cars, but the effort he puts into finding someone who can work on his Ford is not helpful to me if I own an Audi and need a mechanic who works on non-American cars.

A related reason for knowing why interaction history occurred is goal discovery, a form of serendipity. Goal discovery is the process of starting off on one task and realizing that it relates to, or can be co-accomplished with, another task. For example, one could start off to build a system of software agents and realize along the way that this is similar to, and that the tools can be used for, an artificial life system. Goal discovery is important to organizational learning. As Suchman documents, things in office environments almost never happen according to the written procedures. Instead, people find themselves accomplishing tasks by ad hoc and creative responses to changing situations. Knowing that these responses can achieve organizational as well as personal goals is an important part of adaptive behavior. An interaction history system which methodically recorded and made available these discoveries could significantly aid the improvement of processes, products, and services by companies, called “organizational learning.”

Reference 76

Knowing why something happened is crucial for explanation and learning. For example, I may be in the process of making an addition to my house and knowing that the previous owner built up the flooring to avoid rot from poor drainage could be very important. Likewise, software engineers often find themselves concerned with design rationale as they attempt to maintain software systems. Having the code, the comments, and the documentation is often not enough since some questions can only be answered by knowing why a particular approach was chosen, and why other alternatives were rejected, if they were even considered at all.

Orr points to this lack of “why” information as a critical failure mode in directive-style documentation. In this style, users are given a series of steps to perform (usually in a tree-branching style) and led to the correct outcome through these steps. These documents lack “why” information, and so it becomes difficult or impossible to generalize from the specific steps laid out in the manual to an understanding of the problem that would enable the user to cope with situations slightly different from the prescribed one in the manual. Of course, the real world is replete with such variability, a complexity no prescribed set of steps could ever possibly capture.

Reference 54, 55

Knowing how some bit of interaction history was done can be important for issues of naturalness. Objects afford certain ways of interaction; that is, they make some interactions easier and some harder. As noted above, much documentation concentrates on how things get done. This can be particularly helpful for novice users who may find themselves overwhelmed with the plethora of possibilities and need simple assistance to accomplish a goal. The most common place to see this is in software installation systems. These systems give users step-by-step instructions on how to unpack, install, and often customize new software. It is also seen in some on-line help systems; for example, Microsoft Office’s assistant has a “show me” mode in which it will show the user how to select the correct options from menus, how to fill in dialog boxes, and so forth.

Knowing ‘how’

Affordances in the real world are often customized to our bodies. Chairs are for sitting because of the stable surface they give, at approximately the right height. This is

why things like tree stumps, boxes, etc. come to be used as chairs. Likewise, we adapt our artifacts to work together: streets are built to be wide enough for the cars of the day; overpasses are built and rebuilt to accommodate higher trucks and buses which must pass under them, etc. These adaptations and uses provide natural ways for wear to be recorded. We see how things have been done in the past because they leave their traces. Software which was built to show this kind of use information might be termed “wearware.”

A Design Approach to the History Framework

Given these six properties, it is reasonable to wonder how a designer might approach them. In creating a history-rich interactive system, the primary concern is what questions you want your history information to be able to answer. As outlined under the last property above, it is not enough to simply capture users’ interactions without first having a concept of what the captured data will do for your users. For example, if you need accountability, then you must know who did something. If you are trying to teach techniques, you must know how something is done.

The second consideration must be proxemic versus distemic. The fundamental question is whether interaction history can be blended as part of the natural processes of the system or whether dealing with history is to require additional formal rules. Note that simply having formal rules is not by itself a bad thing. Many domains, and systems which assist users in those domains, operate by means of formal rules. If interaction history can be made a part of the rules of the system, it can be much more proxemic than if it requires additional rules. Deliberate distemia may be necessary if issues of privacy and security must be addressed. That is, people may need to be made aware explicitly of how interaction history is being used if their personal data is being exposed.

In making this decision, it is also likely that decisions on active versus passive will be made at the same time, because active history often requires users to operate some different set of controls in order to create it. These controls may be similar to things with which the user is familiar but will likely require some new learning. Conversely, a desire for more proxemic history forms will usually lead designers to consider ways to capture and redisplay history without requiring new learning or disturbing the user’s sense of familiarity and comfort.

The next, related, question is where the history-rich system is being deployed and for whose benefit. A personal system has far fewer privacy concerns, for example. Group systems may choose between social display of the results (for example, on a commonly accessible Web site) and personal displays, such as in the individual users’ Web browsers.

Finally, designers must consider how long history information will be relevant and therefore how (and how quickly) to fade it out. Different contexts may call for different approaches to fading; for example, a system which tracked fads on the Web would probably discard history information quickly. A software maintenance system would probably maintain the history of change information as long as the product was supported by a company.

What Interaction History is Not

We want to differentiate between the area of interaction history and what is conventionally thought of as “history.” The latter involves, insofar as it is concerned with

objects, a preservative activity. Objects taken to be exemplars or individuals,¹ are set aside. This act of preservation deliberately removes the collected object from the sphere of ordinary interchange. The state of the object is considered as a whole and preserved as such, often including the particular set of changes made in the object through its lifetime of use. Undesirable wear is often removed in an attempt to make the object seem more like it did at some point in the past, particularly a point with significance for the object or for an event in which the object participated.

For example, the restoration of the Sistine Chapel's ceiling involved not only an effort to remove the accumulated dirt, but also a repainting in an attempt to restore some of the brightness of color which was present when the original was painted. In fact, this "restoration" is actually creating a new object, painted by artists other than the original, using modern paints, brushes, and techniques. This sort of difference between an authentic object and a restored object may be trivial or it may be crucial, as the Egyptians found out when they attempted to "preserve" the Sphinx with modern materials in the early 1980s. The modern concretes leached corrosive acids which ate away the original stone even faster than before.

Not all such preservation efforts are failures, of course. Museums feature thousands of objects preserved in carefully controlled atmospheres and in arranged glass cases. But it is this very act of preservation that differentiates history from interaction history. The preservation of objects necessarily favors one of the object's many historical states; interaction history concerns itself with the processes that continually mutate objects and our relationships to them. In order to maintain a favored state, any processes which threaten to mutate the object must be terminated.

Nowhere is this conflict felt more strongly than in architecture; this is interesting in part because buildings are the oldest and most commonly found history-rich objects in the world. Brand points out that architecture as it is presently taught is a "fire and forget" profession. Architects view their jobs as done the day the building's first owner or tenant moves in. The changes — the wear — which the owners, tenants and visitors will make to the building are seen as moving it farther from its original idealized form. This is true even when the changes are made to make the building more usable, more livable or to adapt it to a purpose unforeseen by the original designer and builders.

Reference 8

We can see similar tensions in organizational systems corporate offices. New people bring new ideas and new technologies which are introduced into the organization; these often carry with them requirements for whole new practices. Sometimes the organization fails to adapt or properly integrate change: "we've always done it this way" is a sign of an organization in a peculiar form of interaction history paralysis.

*Evolution versus
preservation*

Likewise, we have similar problems with information preservation, particularly in dynamic environments such as the Web. Documents published on the Web may vanish or become unavailable in ways that publications such as print journals never did. More interestingly, authors may continue to evolve documents, perhaps modifying them to reflect feedback from readers, or to incorporate new data and new discoveries. People who wish to make use of these documents must be aware that the pointers they have may be to information that no longer says what they thought it said. Obviously,

1. An exemplar is an example or representative of an everyday and commonplace item, such as a jug used to store grain in Tutankhamon's tomb. An individual is an unique item, usually associated with a person or event, such as a pen used by the signers of the Declaration of Independence.

there is an advantage to being able to refer to a fixed (preserved) form of information; however, to do that is to destroy one of the most interesting aspects of digital information, its mutability.

Even in cases where adaptation takes place, there is an ongoing struggle. On one side is the need to preserve stability, to take advantage of the experience of the existing people, and not create the chaos that comes with too-rapid change. On the other is the need to improve, to integrate new people, ideas, and technologies, and to adapt to ever-changing contexts of action.

Interaction history plays an important role in these struggles, serving as guide, reminder, comfort, and teacher. Unfortunately, our computer systems remain blind to these needs and to the importance of interaction history in general.

2.4 Using interaction history

The processes by which we come to use interaction history are still largely unknown, and yet they pervade our understanding of the world so thoroughly we take them to be normal. For example, let us imagine that we are driving down a highway and we approach a curve in the road. Ahead we see a guard-rail which has a number of black streaks on it. We do not have to stop and ask what these streaks mean; we do not require guides, signs or any other interpretation. We “simply know” that those streaks were produced by the bumpers of cars which went into the curve too fast and so collided with or scraped against the guard-rail. Knowing this, we can slow down and negotiate the curve safely. We take advantage of the work done — the records left — by others so easily we think it is natural.

But neither guard-rails nor car bumpers are “natural” objects. Instead, both are cultural artifacts, bound up with the rituals and practices of our society. Somewhere along the way to becoming literate adults in our society we learned how to “read” such signs and interpret them. Objects “talk” to us, in response to our examinations and uses.

Reference 18

One of the best-known examples of interpretation of interaction history is Arthur Conan Doyle’s famous detective Sherlock Holmes. Doyle’s stories of Holmes are replete with examples of the detective making what seem to other characters to be amazing deductions and having extraordinary insights. However, when queried, Holmes reveals himself to be a master reading interaction history: he takes small cues — such as the presence of ink stains on a person’s fingertips or the type of waistcoat he wears — and draws inferences about the activities, locations or professional life which must have left these traces. For example, the first Holmes story, “A Study in Scarlet,” has him amazing Scotland Yard detectives after observing a crime scene:

Reference 18, page 32

“He was more than six feet high, was in the prime of life, had small feet for his height, wore coarse, square-toed boots and smoked a Trichinopoly cigar. He came here with his victim in a four-wheeled cab, which was drawn by a horse with three old shoes and one new one on his off fore-leg. In all probability the murderer had a florid face, and the finger-nails of his right hand were remarkably long.”

In the pattern we now find familiar, Watson insists on explanations for these assertions. Holmes eventually obliges:

“The very first thing which I observed on arriving there was that a cab had made two ruts with its wheels close to the curb. Now, up to last night, we have had no rain for a week, so that those wheels which left such a deep impression must have been there during the night. There were the marks of the horse’s hoofs, too, the outline of one of which was far more clearly cut than that of the other three, showing that that was a new shoe. [T]he height of a man, in nine cases out of ten, can be told from the length of his stride. [...] When a man writes on a wall, his instinct leads him to write above the level of his own eyes.

“Well, if a man can stride four and a half feet without the smallest effort, he can’t be quite in the sere and yellow. That was the breadth of a puddle on the garden walk which he had evidently walked across. Patent-leather boots had gone round, and Square-toes had hopped over. [...] The writing on the wall was done with a man’s forefinger dipped in blood. My glass allowed me to observe that the plaster was slightly scratched in doing it, which would not have been the case if the man’s nail had been trimmed. I gathered up some scattered ash from the floor. [S]uch an ash is only made by a Trichinopoly.”

Reference 18, pages 32-33

Although some of his methods require a closer-than-normal inspection, and some computation, as of the height of the person from the stride length, what Holmes is doing here is reading the interaction history from the objects in the scene. He combines his observations with logical deductions and some specialized knowledge. From our point of view, the “amazingness” of Holmes’ deductions is reinforced by Doyle’s use of facts we are not shown in advance, such as “no rain for a week.” However, that does not change the basic processes which are going on. In the terms of our framework for interaction history, we would say that Holmes is unusually adept at making interaction history proxemic.

See “Property 1” on page 22.

Aside from authorial trickery, it is worth considering what makes Sherlock Holmes such an insightful reader of interaction history. He observes in detail instead of just superficially, and has studied several seemingly esoteric areas, such as the cigar ash in the example above, which he draws on to fill out details in his analyses. We might characterize Holmes as a particularly expert practitioner, but all of us are skilled practitioners to some degree, at least within our own cultures, as the guardrail example above shows. The question then, is what is the relationship of practical skill to interaction history.

2.5 Skilled practice and interaction history

Donald Schön gives us the notion of a “reflective practitioner.” This is in contrast to the typical notion of what it means to be skilled in an area such as medicine, architecture or other professions. Schön begins by noting that traditional practitioners are supposed to rely on repeated experience and a body of knowledge/expertise. This body includes facts, theories, abstract concepts, analytic methods and examples. When confronted with a situation, the technical practitioner extracts an answer from the body and applies it to the current situation, producing a solution.

Reference 66

Schön points out, though, that most real-world situations are characterized by “complexity, uncertainty, instability, uniqueness, and value conflict.” These situations are typically not covered in conventional descriptions of professional knowledge. Often, as the Sherlock Holmes example shows, the skill is not so much in solving the prob-

Reference 66, page 18

lem as it is in finding the problem details and relevant features in the first place. The technical rationality model isolates the practitioner from the messy world of everyday operations, and shrouds him in what Schön calls strategies of mystery and mastery. This model does not reflect how professionals actually operate. As Schön says:

Reference 66, page 49

“When we go about the spontaneous, intuitive performance of the actions of everyday life, we show ourselves to be knowledgeable in a special way... Our knowing is ordinarily tacit, implicit in our patterns of action and in our feel for the stuff with which we are dealing.”

Schön emphasizes an alternative model of skill, one in which thought and action coexist and interoperate. This reflection — in which thought and practice work upon each other — is the heart of the expertise of the reflective practitioner. Schön suggests a different relationship between theory and practice than is conventionally assumed. Theory is not determinant; rather it is a descriptive.

Reference 66, page 273

“An overarching theory does not give a rule that can be applied to predict or control a particular event, but applies language from which to construct particular descriptions and themes from which to develop particular interpretations.”

That is, theories do not provide answers or solutions; instead, they provide elements (language and themes) from which we can construct problem descriptions. The framing of a problem implies a set of possible solutions. For example if I, as a doctor, frame a patient’s complaint about chronic pain as hypochondria, that suggests a wholly different set of possible solutions than if I frame it as a physiologically based ailment. Schön shows several examples in which an expert practitioner (architect, doctor, and town planner) appear to “solve” problems experienced by novices. These solutions are seen to be the results of framing the problem by the experts in different ways than were thought of by the novices. In our terms, we would say that in order to use interaction history, a practitioner must be reflective.

2.6 Satisficing and interaction history

Reference 66, page 62

Another important part of this reflective practice is that it occurs while we are doing our normal activities. The technical rationality model prescribes thought before action; the reflective model emphasizes reflection-in-action. That is, we reflect as part of normal operation; it is not a special separate activity. Attempting to adhere to this new model has some implications for system design, as we shall see in later chapters of this thesis. Of course, the time available for reflection is bounded by what Schön calls the “action-present” This is, simply put, the time in which action can make a difference. Obviously it does us no good to spend so much time reflecting that we lose our chance to act. In fact, since reflection is inextricably bound up with action, failure to act becomes a kind of dual paralysis.

Reference 71, 72

This is related to Herb Simon’s notion of satisficing. This idea, sometimes also called subjective rationality, is simply the notion that people and organizations do not search for optimum or maximal solutions, contrary to the assumptions of classical economics. Instead, they perform a combination of satisfying and sufficing: satisficing. Part of the reason for this is, as Schön notes, that time is often limited, since we do our reflection while in the course of our intended tasks. Time, of course, is only one possible resource which is limited in real human behavior. Other resources include tangible things, such as money, and intangibles, such as attention.

In addition to the obvious limitations on resources, satisficing also involves limits on human knowledge and energy. We often do not know if the expenditure of additional resources — even a very small amount — will make the solution better or worse. Therefore, we find a level which we deem “good enough” for the particular problem, and usually stop trying to improve things once they have passed this level of goodness. In fact, beyond that level, peoples’ behavior often more resembles randomness than any model of rationality.

For example, I often like to eat out. When I visit new cities, I will usually find an area of the city I’m visiting where a number of good restaurants are close to each other. In such an area, I am usually well-assured that I will get a good meal. Therefore, I do not take a lot of time comparing the restaurants. I may glance at the menus, I may choose randomly, or I may use a simple interaction history heuristic, such as picking the restaurant which has been in business the longest or which has the most people in it at that moment. This is more than a flippant example; it is intended to indicate that interaction history can be a valuable tool for helping people both get up to that “line of satisfaction” and for making quick, useful decisions once that line has been reached. Of course, this depends on the situation and the problem to be solved.

Given this setup, the question then becomes: “How do practitioners satisfice in the course of their everyday activities?” Schön posits a form of conversation that takes place between the practitioner and the people and materials of the situation. This conversation takes the form of an experiment in which the reflective practitioner makes, imagines, or designs an intervention in the situation. This intervention produces a change, a difference which allows the practitioner, and the other people in the situation, to see if they are moving closer to or farther from the desired end state.

How skilled practitioners satisfice

These experiments are not random changes, of course. They are changes based on the practitioner’s repertoire of images, examples, understandings, and past actions. One could say that the difference between an expert and a novice is largely the body of these things on which the practitioner can draw. This is, for example, one of the major conclusions of Orr’s work with photocopier repair personnel. The process is cyclic, of course. The practitioner probes the situation by changing and the situation “talks back” by changing in observable ways. The changed situation is then comparable to the remembered previous situation, in what Schön calls an experiment, in specific ways:

Reference 54

- Can I solve the problem I have set?
- Do I like what I get when I solve this problem?
- Have I made the situation coherent?
- Have I made it congruent with my fundamental values and theories?
- Have I kept inquiry moving?

Reference 66, page 133

The key difference between these kinds of experiments and the kinds of experiments normally conducted by researchers in laboratory settings is that here the practitioner is part of the phenomenon under study. Researchers usually go to great lengths to remove their own influences or biases from experimental situations and to isolate single phenomena for study. In contrast, the practitioner manipulates phenomena which are part of a naturally occurring situation and which cannot be separated from the situation. This is, to use Hutchins’ phrase, cognition in the wild.

Reference 39

Obviously positive answers to all these questions is the best possible outcome, but recall that reflective practice takes place under conditions of complexity and uncer-

tainty. Any given experiment may not produce definitive yes or no answers. Plus, the situation likely includes instability such that what seemed a good experiment may no longer be relevant. For example, a doctor may advise aspirin and fluids for someone suffering from a flu. If the sickness worsens and requires hospitalization, the experiment of aspirin may be abandoned as no longer useful; however, the need for fluids may increase to the point where they may be administered intravenously rather than waiting for the patient to be able to keep down a drink of water.

There is one major lack in Schön's description: he does not provide a description of how the reflective conversation takes place. In fact, as noted above, there are two such conversations — one between the practitioner and the people involved in that situation, and one between the practitioner and the materials. We will consider in turn how each of these could proceed and the importance of interaction history in this process.

2.7 The first reflective conversation

Schön points to the first conversation as one of the limits of reflective practice. As practitioners, we develop specialized languages which may not be shared with others in the situation. In addition, our interventions and goals may not be understood or shared by other people. For example a patient being treated by a doctor — even one who follows the reflective instead of the technical rationalist model — may not understand the terms used by the doctor. The patient may also not understand why certain treatments are prescribed and may become frustrated with the doctor's inability to explain the situation. Such frustrations may cause the professional to play the role of rationalist expert, even while continuing a reflective conversation in real life.

Reference 76

Other theorists have attempted to explain how such a conversation could take place. In particular, Lucy Suchman in her book *Plans and Situated Actions* provides key insights into the relationship of conversation and planning. She points out that retrospective analyses of conversations or accounts of past activities usually omit particular crucial details, such as errors, omissions, or deviations from plans. Just as with Schank's storytellers discussed below, Suchman found that her office professionals produced idealized accounts of their activities and conversations. In this respect, written instructions can be seen not as prescriptive plans — do this or that — but as idealized descriptions of what was actually done.

Shared context

In part, Suchman points out, this is due to the degree of shared assumptions and background thought to exist among the conversation participants. We say less if we feel we have more shared context with our listeners. Conversely, saying less — not spelling out all the details — leads to more assumed shared context. This can break down quite easily, of course. It may seem obvious to a speaker that something should be done or did happen and, assuming that hearers share this context, the obvious is left out. However, if it is not part of the shared context, it will not be obvious and may lead to confusion, or may end up with the "obvious" not being done.

This shared context is in part cultural and in part a function of one's expertise — that is, one's degree of skill — in the particular job. Thus, someone who is a skilled professional in one context may find himself at a significant disadvantage when in a slightly different context. For example, a surgeon from a hospital in the UK may find herself at a loss when visiting a US hospital not because her skills are in any way less than those of her American counterparts but because she lacks the necessary shared background. Crucially, her colleagues assume that she has this information and so leave out things that might be thought to be insultingly obvious, yet are important enough that without them one can appear clumsy or uninformed. In some high-pres-

sure cases, such as protocol in the operating room, these “obvious” details can seriously hamper the ability of the team to function.

Suchman focuses on the difficulty of inferring what others are trying to do, based on their observed utterances and actions, stated intentions and circumstances. Of these three, she argues that the first two — what we intentionally communicate — are insufficient. Only the third, the observable and inferable surroundings, is deterministic:

“[T]he prescriptive significance of intentions for situated action is inherently vague. The coherence of situated action is tied in essential ways not to individual predispositions or conventional rules but to local interactions contingent on the actor’s particular circumstances.”

Reference 76, pages 27-28

In light of this, we must look to other things for an explanation of what others are doing. Clearly though we do attribute plans to people, as Suchman points out, plan attribution is not the same as plan recognition. In fact, the notion of plans in general is inadequate to explain how we — to insert Schön’s term — have a reflective conversation with our co-participants.

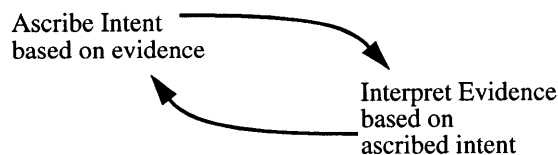
Plan attribution

Suchman’s move is to re-cast plans as both representations of actions and as ways we orient ourselves for situated actions. That is, plans form the same kind of convenient, detail-omitting shorthand as was described above; plans codify those details which skilled practitioners might not be expected to know or to have already in mind. This is one reason why professional plans, such as office procedure manuals, transfer so poorly from one context to another. No doubt we have all heard stories of the “culture clashes” that occur when two companies merge. Each comes with its own set of understandings and the external representations — the corporate planning and procedural documents — are inadequate to convey how things actually get done.

Reference 83

Notwithstanding the understood inadequacy of plans as coordination mechanisms, Suchman points out that we do still use plans. In particular, she says, we use them to orient ourselves toward a subset of the very large set of all possible things which might happen. In essence, we operate in a tight reciprocal loop, as shown below:

FIGURE 3. Using plans in real conversation



However, any retrospective account of this loop is difficult because, as Suchman found by analyzing videotapes of her subjects, “fleeting circumstances” were used systematically in interpretations of actions, but were inevitably omitted from post-hoc descriptions.

Reference 76, page 109

In a completely planless environment, action might be chaotic, with no logical order at all. However, this is not how skilled practitioners proceed. In fact, their speech and actions are usually purposeful — as Schön says, they are oriented toward interventions which should change the present situation into one closer to what they desire.

Plans are a satisficing technology

Instead of chaos, we use plans as a form of satisficing technology. The plan is not a precise descriptor, nor always an accurate predictor, but it is usually good enough. This is particularly the case if the person is willing and able to dynamically or opportunistically re-plan. In fact, it is hardly controversial to say that one of the hallmarks of a skilled reflective practitioner is the ability to adapt well. This adaptation means not being put completely off-balance by new situations, it also means being able to change one's goals, purposes and course of action to suit new situations as they arise.

Therefore, we do not look to rules and procedures — the components of plans — for self-contained or foundational information. Instead, we see that they are both contingent on and are determined from situated actions. Simply put, if we see a coworker clean his tools before use, we might infer a rule that says “tools should be cleaned before use.” However, seeing a tool used without first being cleaned does not necessarily invalidate the rule. Instead, it may offer an insight; for example, we may learn that time can be saved by cleaning tools before they are put away. Suchman points out that much organizational learning depends on such opportunistic insights.

2.8 Interaction History as the Second Conversation

Suchman has given us the framework for understanding how the first reflective conversation — that between the practitioner and the people in the situation can happen. I propose that interaction history is the mechanism — or one of the mechanisms — by which the second conversation — that between the reflective practitioner and the materials of the situation — takes place. That is, practitioners learn to “read” interaction history in somewhat the same way that Sherlock Holmes has learned to read a crime scene. The objects in their areas of expertise reveal to skilled practitioners information that is obvious if you know what you're looking at. Of course, the unskilled practitioner (customer, client, patient, etc) cannot read these cues until they are pointed out; in retrospect the obvious can be made clear. Specifically, what must happen is that we read these cues not as incoherent or isolated entities, but rather as important elements in the creation of a coherent story. Interaction history is usable by a reflective practitioner only to the extent that a coherent narrative can be made out of the traces.

Reference 64, 65

Roger Schank argues that stories form the basis of human memory. In particular according to this theory, stories form the basis for conversations and for structuring memory internally and, finally, enable us to understand new information by placing it in a coherent or ordered relationship to information we already know. Stories provide a unique structuring form for understanding information by allowing us to vary disparate elements such as character, setting, or time. This variability enables us

- to form and mentally test hypotheses about the future state of the world;
- to see similarities and differences (compare and contrast);
- and to explore alternative interpretations not only of events in the stories, but of events in our lives.

These three activities are central to our claims about the usefulness of interaction history.

The car accident story

One particular illustration of this theory is worth considering in more detail. Schank reports on a series of experiments in which subjects were asked to imagine that a simple event had taken place. In taking their mother's car to get gas, they had driven into and broken a clay planter which the gas station owner had placed next to the gas pumps. Initially, subjects produced a dry retelling for the experimenter; this retelling was quite similar from subject to subject. Schank then has the subjects retell the

story, but instead of telling it to the experimenter, they were asked to imagine that they were telling the story to different people. For example, they were to imagine that they were telling it to their mother, to a police officer called by the station owner, or to a friend a month later.

In these conditions, the stories told by the subjects varied widely. Subjects “embellished” event, added information, left out information, changed emphases, and so on. Schank shows that subjects were clearly not just relating a series of events; they were telling a story. That is, they were giving a purposive interpretation. The story did not merely involve recall of events and descriptions; instead, the story was created anew for each telling, and the perceived (or imagined) context for that telling provided a guiding structure on which the tellers built their narratives. Schank has argued that we use stories to make sense of facts, to put them in context. As part of our process of understanding, we construct; by constructing, we understand.

For example, Figure 4 shows a track in Boston’s underground Park Street Station. The problem passengers have is to figure out where along the track they should stand in order to get onto the train once it arrives, since there are no signs indicating “board here” or “train car stops here.” One possibility would be to use a form of social navigation and ask other passengers waiting for the same train. Alternatively, one could seek the answer in the environment itself.

FIGURE 4. Park Street Station



In particular, as show in Figure 5, there is a safety border painted on the sides of the track. Unless newly repainted, that border clearly shows signs of wear, a form of interaction history. In order to make sense of this observation, though, we must make a story for ourselves with elements of how trains pull into the station, how they stop in approximately the same place, how people step onto and off of the train in particular places, and so on. In constructing this story we come to understand, to make sense of

our observations, and we do it so easily that it seems natural to us. We don't realize we are doing it until something happens that disagrees with our interpretations.

FIGURE 5. Wear in the station (floor closeup)



2.9 Social Memory

Reference 19, 20, 3

Memory is much more than individual recall, and interaction history is intimately tied in to what is called *social memory*, *community memory*, or *collective remembering*. These terms come from a school of thought beginning with French sociologist Emile Durkheim and British psychologist Frederick Bartlett at the end of the nineteenth century and carried on today in several areas such as psychology, anthropology, history, and sociology. In this view, remembering and forgetting are social processes, not simply located within the head of a single individual.

For the purposes of this thesis we are not concerned so much with the abstract processes of memory as we are with the relationships between interaction history, history-rich objects, and the memory-related means by which humans make use of these objects. As Radley put it:

Reference 59, p. 49

It is not just that individuals remember specific things, or are reminded of the past by particular objects figured against the background of a shared discussion of the past. Artefacts and the fabricated environment are also there as a tangible expression of the basis from which one remembers, the material aspect of the setting which justifies the memories so constructed.

Memory is part of an exchange

That is, memory for our purposes exists in the midst of an ongoing exchange between social custom, artifact, environment, and history-rich or history-preserving structures. Important questions which arise from this field of study include those about the nature of remembering together, the social practice of commemoration, the social foundation and context of individual memory, the rhetorical organization of remembering and forgetting, and the nature of institutional remembering and forgetting.

Remembering together

In remembering together, two kinds of memory dominate. The first is a simpler one in which people share memories of events and objects that are social in origin. In

fact, we take such shared memories as a necessary condition for the formation of communities. We all remember where we were when the Space Shuttle blew up or, if we are older, where we were when President Nixon resigned or when President Kennedy was shot, and so on. Even if we did not personally participate in these memories, we share them as a kind of cultural reaction to an essentially social event.

More complexly, though, there is also a sense in which a group of people can collectively form and function as an integrated memory system. In this form of memory, the whole of the recollection is more than the sum of the individual perspectives. Wertsch showed that recounted shared experiences became reinterpreted and recounted by other storytellers in later tellings. That is, in his experiments, people who heard stories told by others about events which they themselves remembered later incorporated elements of others' tellings into their own memories and retellings and could not afterward reliably distinguish the two sources. Hutchins also showed that groups could rediscover how to achieve some end which no one of them actually had the knowledge to achieve independently.

Reference 79

Reference 38

Commemoration involves an object, person, or event achieving historical significance. This significance is acknowledged and remembered within a community. Again, it is easy to see that agreement with the particular cultural interpretation of the significance is a sign of community membership. That is, if you and I assign the same or similar significance to a particular commemorative object, we are likely members of the same community, with respect to that object. Often the practice of commemoration involved the recall of communally significant elements, even if those elements have no personal significance.

The social practice of commemoration

For example, the Vietnam Veterans' Memorial in Washington, D.C. might be felt to be significant by an American who fought in that war. By contrast, an American who had not fought in the war and who did not find the Memorial significant would thereby show himself to be excluded from that community even though an outside observer might naively assume the two Americans to be communally related.

In setting a social context for individual memory, we are asserting that what it means to recall, remember, and forget are social and culturally determined processes. In this view, memory is a product of social dynamics, not a given. Therefore, it is unsurprising that what is considered a history-rich object in one culture may be considered irrelevant in another. We must all learn (be taught) what things are important to remember. Anyone who has ever listened to a small child tell a story of what happened to him sometime in the past can realize this. The child includes enormous numbers of things which we as adults realize are not significant. But no one ever writes down rules for what is "correct" to remember and relate. As the discussion of plans and procedures in this chapter shows, it is some of the most important things which go unwritten in preparing formal instructions.

Social context for individual memory

Where commemoration often involves strong elements of standardized ritual and rote performance — such as singing the national anthem before a sporting event — social remembering, such as conversation, is much more free-form and open-ended.

Social remembering is free-form

Memory in the real world — or cognition in the wild, as Hutchins calls it — is not done on its own. It is an act performed in a social context, under certain promptings, and with certain goals in mind. Often people required to remember are placed in a contradictory position where their relationship to the remembered events is not necessarily positive. Thus the recall becomes not only about the events but about the teller as well. We may omit details of errors, missteps, false starts and so on in our plan

description not only because we have forgotten them or deem them unimportant, but because we do not wish to appear bad to an audience or those who will later read our written accounts.

We have strong incentive to produce idealized accounts. It is no accident that “Don’t kill the messenger” is still a popular aphorism in our culture. It is also no accident that those in power — from Orwell’s Big Brother to real-world national governments — have sought to control the present and the future by revising the official accounts of the past. History-rich objects are intensely valuable in such situations because they can serve as the only validations for a personal account which in some way contradicts or differs from the accepted or imposed social constructions of memory and history. The records left in an object are in many ways seen as more objective than the recall of individuals. In particular, since each person reading the interaction history must perform his or her own story-creating in order to make sense of the traces, history-rich objects enlist the audience to our cause. We bring in the attentive and cognitive wills of the other story tellers as they must actively make the historical traces sensible rather than being handed a preconcerted explanation by another party.

2.10 Organizational Memory

Reference 39

Ultimately, there is a sense in which all organization — nations being only one large-scale example — can be said to remember and, thus, to forget. Organizational learning is a blend of individuals learning or remembering, and the embedding of knowledge into the tools, techniques and practices of the organization. Hutchins gives a detailed account of the encoding process as it relates to the organization of a modern sea-going ship’s navigation team. For example, one of the key activities of navigation in these environments is “shooting a fix.” In this activity, a seaman uses a device called an alidade through which he sees a thin vertical line. This line is aligned with a stationary object (shooting) at a given time (the mark). The seaman then reads off a number from the alidade and reports it to the quartermaster, who enters it into a log.

The alidade provides the number for the angle of the shooting directly to the seaman because it is integrated with the ship’s gyrocompass. In effect, the device and the organizational system it is part of have “learned” to make the transformation between the line of the alidade and the angular number. In the past, this transformation was made by hand via a geometric computation or lookup. Indeed, the same learning could be embedded in the head of a person (computation) or in a look-up table which could be passed from person to person. In each of these arrangements, the organization which is the navigation team “remembers” how to accomplish the transformation whenever the task is done. One of the advantages of the current form of representation is that the knowledge is encoded such that, if a person leaves the organization, the functioning of the organization is not usually hampered, though new seamen do need to be trained to use the alidade as part of the navigation procedure.

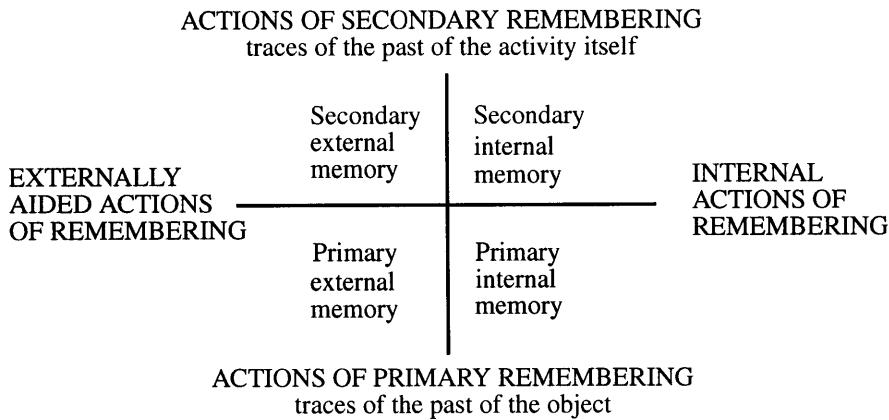
The disadvantage of encoding a procedure within a representation or an object is that the procedure may not be otherwise available when it is needed. For example, many of us have seen store checkout clerks who could not properly make change when the computerized register was not working and not precomputing the change for them. Similarly, important information can be lost — and organizations can be said to forget — when history-rich objects are worn out or replaced. This may be individual, for example when a new tool handle causes blisters because it lacks the grooves worn into the old handle. Or it may be social, for example when a well-thumbed reference work is replaced by a new copy that lacks important modifications added by many people after many months. The notion of “forgetting” is analyzed in more detail in the description of the “Fade Function.”

See “Property 3” on page 23.

Engeström et al. propose a method for thinking about how organizations remember. Their framework, reproduced in Figure 6, emphasizes two orthogonal properties. The first is external versus internal remembering, and deals with whether the remembering involves a consciously created reminder/mnemonic or not. For example, tying a string around one's finger involves an external aid to memory or using a mnemonic list to remember the names of people you have met is an internal action.

Reference 21

FIGURE 6. Framework for analyzing remembering in work (from Engeström)



The distinction between primary and secondary is a way of differentiating the memory of the past from the creation of memory for the future. Remembering “how it was done before” is primary remembering; writing down the present set of procedures and plans is secondary remembering. Primary memory is what is necessary for individuals to operate in the present; secondary memory is needed for future operation. This distinction is similar to the active/passive property in our framework below. History-rich objects figure prominently in Engeström’s organization of memory. A differentiation is made between history information associated with the past of the activity, which is primary for them, and the history of the object, which is associated with secondary memory.

Primary memory is for present operation; secondary memory is for the future.

Engeström and his coworkers make the important point that within an organization, or “social interaction history” as it is called below, we must remember that not all participants have equal access to the memories that are part of remembering. Even if a performance of a procedure is open to observation, not all observers will see the same things. In addition, the objects associated with external memory are often both highly idiosyncratic and personal. This is an important point for designers of history-rich systems to keep in mind. The conditions under which history-rich objects are shared, if at all, will be critical.

Reference 21

Objects, particularly history-rich objects, play critical roles in each of these forms of social memory. We see objects not only as triggers or as mnemonics, but as active participants in the process of “re-remembering” — that is, the putting-together-again of the coherent stories which make memories useful. Objects clearly act as triggers, in that they inspire the start of the remembering process, but they do not then vanish from the scene. We often hold or touch or look at the object as we recollect. Elements or features of the object continue to inspire or encourage us. In this sense, the object is also mnemonic in that it encodes a representation of the memory trace. However, if it was only a mnemonic, any object of the proper category could serve to elicit the recall. In reality, though, we are very particular about it being precisely the right object: this is my grandfather’s axe, not just any axe. History-rich objects are as

Objects’ role in memory

important to our social remembering as are human conversational participants — as anyone who has found himself having a “conversation” with a treasured picture or relic can understand.

This also is evident in the way we choose to leave certain history-rich objects to inheritors. We do not simply parcel items out arbitrarily; we give with care, in part because it is our memories — bits of our selves — that we are giving away. With these bits we seek to create in others the image of us that we wish them to have, just as the stories related by Schank’s and Suchman’s subjects were slanted toward creating desired impressions in the listeners. Once we are gone, we can no longer speak to others, so our objects must speak for us.

2.11 Summary

This chapter has done two things. First, we have provided a theoretical definition of and framework for interaction history. The framework takes the form of six properties which can be used to describe any possible history-rich system. Second, we established this new framework in terms of existing theoretical work, particularly Schön’s notion of reflective practice. In particular, we argue that reflective practice requires three things: first, conversational action as described by Suchman; second, the reading and creation of interaction history stories and their integration into narrative memory as described by Schank; finally, an understanding of the ways in which objects and memory are co-dependent, particularly when memory is considered as a social phenomenon. Our goal in this chapter has been to illuminate how interaction history operates in the physical world and suggest concerns and guidelines to which designers of digital system should attend. Next we turn to implementations of these ideas in digital domains.

CHAPTER 3*Navigation and
Navigation Tools*

*Do not go where the path may lead, go instead where
there is no path and leave a trail.
Ralph Waldo Emerson*

Given the theory and framework of the preceding chapter, there are a number of areas for straightforward applications in which to test the theory. Areas such as electronic commerce, knowledge management, and organizational learning are all obvious possibilities. For this thesis, we have chosen to focus on the problem of information navigation, particularly navigation in large, heterogeneous, complex information spaces. From the users' side we consider the problem of how to get around in such a space; from the designers' side we consider the problem of how to organize a space so that users can get around. The most well-known complex information space today is the World Wide Web, and indeed our tools run over the Web. However, our domain of interest is navigation in all such spaces, which include databases, knowledge repositories, and even physical spaces such as libraries, retail stores, and museums. Each of these presents a particular arrangement of information and challenges users to find their way around.

We will begin this chapter with an examination of the most commonly used model of what navigation means — a model that considers navigation as a simple wayfinding task. We then propose an alternative model, which considers navigation as a foraging activity. We adopt Lynch's 5-element framework for describing peoples' cognitive structures for arranging information as a way of thinking about what structures users might envision. Our structures are similar to Lynch's. Under this framework we present our Footprints tools: maps, paths, annotations, comments, and purposes. Each tool is described in detail. Finally, we summarize our development in terms of the framework of Chapter 2.

Reference 44

The focus on navigation comes from my own long-standing interest in navigation problems rejuvenated by Hutchins' account of how navigation takes place on board modern ships and of the roles played in this navigation process by history-rich artifacts.

3.1 Motivation

References 23, 39

Following Roger Downs and David Stea, navigation is usually described as a four-part activity:

Reference 17

Downs & Stea's 4-part navigation model

1. Orienting oneself in the environment
2. Choosing the correct route
3. Monitoring the route
4. Recognizing that the destination has been reached

Reference 39

However, this model has several problems. First, it is an individualistic notion of navigation that omits any sense of the cooperation and coordination necessary for real-world navigation. Although more complex navigation models — such as Hutchins' — can be said to be captured by this four-step process, it is clear that this simplified model omits many of the key complexities of the navigation task.

See "Social Memory" on page 40.

Second, the model is oriented toward navigation as a wayfinding within the head of one person. This is contrary to the research and theories presented in Chapter 2, where we discussed the social nature of group tasks, of skilled practice, and even of memory itself.

References 13, 31

Lastly, the model assumes that not only is the destination known at the outset, but also the route needed to reach that destination and the spatial relationship between the navigator's present location and the target location. In most navigation situations only some (or none) of these conditions are true. This is particularly the case in digital information spaces, as has been known in the hypertext research community since at least the late 1980s (Conklin 1987, Halasz 1988).

3.2 Alternative Model of Navigation

If, instead, we apply interaction-history related models to this task, navigation can be seen as a kind of skilled practice. We acknowledge that it contains essential — yet often overlooked — trial and error, that it is a social process taking place partly outside the confines of any one person's head, and that it critically involves artifacts. These beliefs guide my entire approach to navigation and to the solutions envisioned for navigational assistance.

References 56, 57

The first question to be answered is: if wayfinding as described by Downs and Stea does not cover the range of tasks in which a navigator engages, what does? We will use the model of foraging theory, particularly as applied to information systems by Pirolli and Card at Xerox PARC. In this theory, navigation in an environment serves two purposes: exploration and exploitation.

Exploration

Exploration is the process of becoming familiar with, or learning, the environment. Significant features are learned so they can be later recognized and navigational aids are noted. Sources of possible value are discovered, by either methodical approaches or serendipity.

Exploitation

Exploitation is the process of using things for benefit that were found during exploration. This may be as simple as recognizing a previously seen landmark and using that recognition to find one's way home. Or it may be as complex as making one's way from home to a previously mapped resource, acquiring that resource (food, water, materials, information, etc.) and returning to one's starting point.

Differences from classical foraging theory

Reference 75

Our application of foraging theory differs from classical (biological) foraging theories in several ways. Classical theories such as that of Stephens and Krebs impose several fairly strong constraints. The first of these is optimality. In biological foraging theories, it is generally assumed that animals pursue optimal foraging strategies either by dint of intelligence or by the result of evolutionary pressures. Evidence for this

assumption, though, is often lacking. Classical theories work on a “choice principle” of maximization; that is, they assume that the forager will maximize its gain while minimizing the amount of energy expended. As discussed in the previous chapter, we do not believe humans engaged in expert practice optimize. Instead, they satisfice. We believe that information navigators also satisfice rather than optimize, showing opportunistic and unpredictable behaviors rather than using rigorous methods.

Classical theories also use a form of “currency” to describe the actions of the forager. The units of currency can be food, water or energy — essentially any resource that the animal needs to survive. In an information foraging situation it is harder to say what the units of currency might be. For our experiment, we chose the number of alternatives generated during an exploration task. Other appropriate measures might be the degree to which the information space is covered, or the depth of understanding gained.

Classical theories assert that exploration always precedes exploitation and that the two do not happen simultaneously. Although it is fairly obvious that the chances of successful exploitation are increased by previous exploration, that is not by itself sufficient evidence that activities always happen in sequence. In information foraging we believe they more often occur at the same time. As noted above, resource discovery may be serendipitous, and we may choose to take advantage of the resource at that moment. One of the great delights of browsing a new store is the unexpected discovery — that new CD or long-sought book — that we quickly snatch up.

Foraging theories often assume that the creature does not know its environment; in the situations in which we are interested this is rarely true. For example, we know that bookstores carry books, even in countries where we cannot read the language of the books. Likewise, exploitation may contain within it elements of exploration. We may try new routes when old familiar ways are blocked, or we may notice new things when following our usual ways. Environments rarely stay static and change may force us to adapt in ways over which we have little or no control, possibly by forcing us into a continual process of exploration in our attempt to cope with the mutability of our surroundings. Alternatively, we may be forced out of our normal routes and patterns by change.

In summary, the model of information foraging presented here make a number of significant changes from classical foraging models. We relax constraints on optimality, on exclusivity of action, and on currency of use. Even with these differences, a foraging notion seems to cover more of the activities we want to talk about than the wayfinding model. Within this larger framework, the wayfinding description of Downs and Stea can be thought of as a subactivity of navigation. Each of exploration and exploitation involves a number of wayfindings. These are often dynamic, as goals and subgoals change during the course of the navigation.

Lynch gives us a larger-scale model of how navigation happens. He identifies five major elements formed in the minds of navigators of large physical-world spaces such as cities. These elements are: *paths*, *edges*, *districts*, *nodes*, and *landmarks*. These elements make up peoples’ mental maps and are used for both explorative and exploitative activities.

Paths represent courses or routes that are customarily followed. Lynch refers to them as channels, in acknowledgment of their effect in organizing and directing flow. We follow paths not because we are required to do so, but because the path provides

3.3 Lynch’s Elements of Navigation

Reference 44

Paths

See “Trails in Footprints” on page 63.

arrangement and relationship among elements that would otherwise be incoherent or disconnected. Paths are a key data element in our system, as described below.

Edges

Edges are the dual of paths. They are, like paths, linear sequences of features or sites, but they do not afford access to many destinations. Edges serve to bound areas and often to block paths that would penetrate them. Many edges are features that would require special modification to penetrate: shorelines, railroad tracks, wall of developed buildings, and so on. This leads to them being seen as barriers. Other edges are more subtle and relate to district boundaries. For example, in many cities ethnic areas such as a “Chinatown” will have street and store signs in both English and another language such as Mandarin. The presence of such signs serves to create an edge even when no physical barrier exists. In the World Wide Web, edges appear in many places because links are one-way. It is often hard to backtrack one’s steps, though browser software alleviates this to some degree.

Districts

Districts are sections of the terrain that cause people to experience an inside of/outside of dichotomy. Ethnic districts, as noted above, are one common example of this. We also see it in heterogeneous spaces like the Web where identities transcend Web server locations. For example, the MIT Media Lab “Web site” is in fact served from more than half a dozen machines, each of which serves a different set of pages. Nevertheless, the Lab pages as a whole can be seen as a “district” in Web space.

Districts may also be defined by a texture or feel of interaction. Developers of Web “malls” (groupings of commercial Web sites) have attempted to give this kind of feel to their sites, as have most of the major Web-space providers. For example, Geocities (www.geocities.com) superimposes a “floating” graphic on all pages within their site. This graphic — a translucent copy of their logo — maintains a position at the lower right of any page within the Geocities “district” as the page is displayed on users’ Web browsers. Thus no matter where one goes in the Geocities site, one is constantly reminded of the district.

Nodes

Nodes are strategic spots or focus points. They serve as junctions between points on paths, as entry points into districts, and so on. In the physical world, a node is often a meeting place such as a street corner or office water cooler. In information spaces, nodes are less commonly places cooccupied by people, unless the system allows simultaneous awareness; that is, if the site supports a chat facility or other means of users communicating their presence and conversation. More often, nodes in information spaces are places to which users return for valuable information or to help orient themselves. The most common occurrence of this nowadays is the Web bookmarks that browser tools allow users to set.

See “History Patterns” on page 71.

To the dismay of Web site designers, users frequently set bookmarks to locations “inside” the district of a Web site. Designers often want users to navigate their information presentation in particular ways, which is frequently incongruous with how users perceive that the information should be navigated. This is covered below in more detail, as we saw interesting examples of this in how people used our tools.

Nodes are related to paths, since nodes are frequently the places where multiple paths intersect. Nodes are also related to districts in that the nodes can be entry points to districts or they can be focal centers for districts.

Landmarks

Landmarks are another form of reference point. Lynch says that the difference between a landmark and node is that a landmark is someplace that the navigator remains external to, while a node is someplace that is entered into. In the information

world, landmarks are rare or nonexistent. Nodes must most often be used for landmarks.

Landmarks are things that can be singled out from the environmental background. They are usually visually distinct from their surroundings either visually or in terms of their use. For example, a particular storefront may serve as a landmark even though it can only be seen from very close by, while a water tower or other tall structure may be a landmark precisely because it can be seen for miles around.

All of these categories are fluid and depend on the navigation needs of the person at the moment. As Lynch puts it:

“The image of a given physical reality may occasionally shift its type with different circumstances of viewing. Thus an expressway may be a path for the driver and an edge for the pedestrian. Or a central area may be a district when a city is organized on a medium scale and a node when the entire metropolitan area is considered.”

Navigation elements do not always map uniquely to one of Lynch’s categories. As noted above, districts are bounded by edges. A place may be on many paths. Even so, these categories help us think about tools for navigators.

There are three major ways in which navigation can be said to be related to information systems design and use. First, navigation may be thought of as a metaphor for user practice. As navigation concerns movement from one location to another, users can be said to “navigate” within a system. For example, users may navigate the set of commands and options offered by a program. This is particularly true if these commands are presented as a set of hierarchical menus in which movement of the cursor is directly equivalent to selecting a different command or subcommand. Similarly, popular PC interfaces such as Microsoft Windows and Apple’s Finder offer the ability to “navigate” a hierarchical file system.

Secondly, navigation may be seen as a way to help users with information retrieval tasks. Instead of presenting users with a command-line system full of unusual commands (expecting users to remember those commands) it is possible to present information in a more directly manipulable form. Two- and three-dimensional representations have been used for many kinds of information from organizational charts, such as those produced by George Robertson at Xerox PARC to numerous products oriented towards visualizing the World Wide Web.

The difference between information retrieval and information navigation is small but significant. It is roughly the same as the distinction made above between exploration and exploitation in foraging behavior. Information retrieval is an exploitation activity — its goal is to go out, find a result, and return with that information. Criteria such as time and effort needed to achieve the result are relevant. Traditionally, information retrieval systems are rated in terms of recall and precision. Recall is the percentage of relevant items returned, typically in response to a query. Precision is the percentage of returned responses that are relevant to the query. Salton presents an extensive discussion of information retrieval, precision, and recall.

Context is often ignored in information retrieval, possibly because it is hard to measure. Some systems allow users to do what is called “query refinement” that involves modifying an existing query and rerunning it, rather than constructing a new query.

Reference 44, page 48

3.4 Relationship of Navigation & Design

Reference 62

Information Retrieval vs. Information Navigation

Reference 63

Reference 58

Such systems may restrict the refined query to operation over a set of documents retrieved by a previous query. This set of documents, rather than the entire information space, is taken to be the context of the subsequent query. Pirolli and colleagues' work on Scatter/Gather browsing at Xerox PARC is an example of this.

In this analogy, information navigation is more like exploration. As noted above, this means that there may be a high degree of uncertainty with a high degree of opportunity or serendipity, and so on. It also crucially includes the notion of becoming familiar with the environment during the navigation activity.

In contrast to retrieval, navigation is not concerned with the acquisition of an answer. Speed of reaching a destination is important, but must be balanced against the need for the person doing the navigation to be able to understand how the objective was reached. Measures such as precision and recall are hard to apply since navigation is about a process and a destination for that process, rather than about a set of results.

Reference 40

Context also plays a key role in information navigation. Environmental cues give users ways to monitor their progress, to recognize nodes, edges, and so on. User context also plays a significant role in navigation; task-focussed navigators may simply want to reach their goal as quickly as possible while learners may wish to explore the environment more completely. Interactive information retrieval systems try to include this notion, sometimes called utility, via relevance feedback (see Koenemann and Belkin's work, for example). However, this requires users to evaluate and react to the goodness of search results. In navigation activities, irrelevant results are simply ignored; interesting results capture users' attention.

Reference 44

Another difference is that navigation is as often concerned with the context of the answer as it is with the content. The presentation and organization of an information space to be navigated are usually intended to give cues to the navigator by creating a logic that guides people from one item to the next. As Lynch notes these cues are used by people to form the basic elements of their mental images. In the case of the Web, some of the cues are created from the data structures — the pages and links between them. Other cues need to be added, usually by inclusion of navigation aids such as maps or toolbars. Navigators must then cope with the inconsistency of these cues from Web site to Web site.

Logic of Navigation Cues

The logic of these navigation cues may be spatial, so that our 2D and 3D navigation skills are directly used, or it may be semantic — such as a hierarchy, classification system, or semantic network. In these cases we need another layer of interpretation, one that maps the moves in the semantic structures to navigational moves. For example, hierarchies are often represented as trees, but the trees may begin at the top (left, in the US and other left-to-right language cultures) and proceed downward as things get more specific; or, they may begin at the bottom and progress upward. These two forms are shown below in Figure 7.

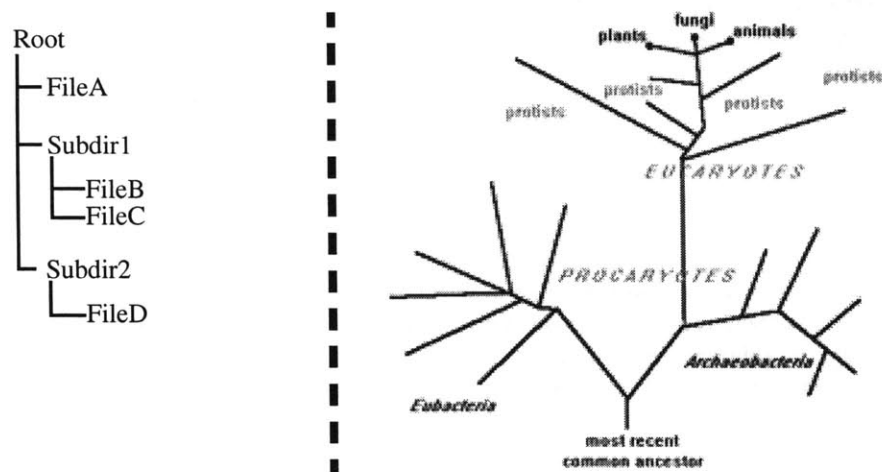
Reference 42

Indirect mappings are not necessarily worse than direct mappings. As Lakoff and Johnson point out, the act of mapping conceptual structures to spatial relationships is natural and occurs in all language cultures. In handling metaphors in speech we easily align concepts such as “healthy” with directions such as “up.” This mapping seems to work particularly well — if existing language structures are any indication — when the navigational mappings are body-centric. That is, the road is “ahead (of us).” However, as Hutchins shows (particularly in his discussion of “Navigation as Computation”) we routinely project our “selves” or our viewpoints to locations that may not exist, or that exist but will never be used as an actual vantage point. For

Reference 39, page 49

example, the viewpoint necessary to use a standard western-style map is that of a bird. In fact, the height needed in order to see the terrain as presented by most maps is one that would be impossible to attain. Similarly, Hutchins shows that Micronesian canoe navigators make use of “islands” that do not in fact exist, but are assigned consistent (imaginary) positions so that they can be used as reference points in a journey.

FIGURE 7. Top-down (left) and Bottom-up (right) hierarchies



Given that navigation requires such mental gymnastics it is surprising that so many of our navigation activities are successful. In fact, research by Nils Dahlbäck shows a difference of as much as a factor of twenty in peoples’ navigation skills. That is, the worst navigators in his experiments were twenty times as likely to get lost, or took up to twenty times as long as the best navigators to complete a navigation task. Thus it is no surprise that we often turn to other people for assistance in navigation. This navigating by means of others’ information is what Dourish and Chalmers called *social navigation*.

3.5 Social Navigation

Reference 14

Reference 16

As with interaction history, social navigation originates in physical world phenomena. We are all used to asking for directions. Usually we try to ask someone we think knows the answer — a friend, a police officer, or someone who seems to be native to the region. This paradigm applies to more than navigation, of course. Social navigation terminology is commonly applied to describe *recommender systems*. That is, we think of the process of recommending — a movie, professional services, etc. — as one of helping others navigate through the maze of possible alternative choices.

Reference 61

In fact, this is close to the original sense in which Dourish and Chalmers talked about social navigation. Their example was the difficulty Web surfers have in finding reliable information on any given topic. The problem has two components. The first is the well-known difficulty of finding something useful or desirable in a large, complex heterogeneous information space. Search engines generally produce far more results than can be reasonably viewed. One approach, primarily associated with the Yahoo! search engine, is to impose some level of external order on the chaos, primarily by means of a categorization, such as Yahoo!’s ontology. Efforts such as these are largely hierarchical in that they assign pages to particular categories, though pages may be in multiple categories. For example, a page on Perl utilities may appear in *Computers:Languages:Perl* and *Computer:Software:Utilities*. These efforts are largely centralized. Although page authors can suggest categories into which they think pages

should be put, the final authority rests with the organizers of the ontology (“ontologists,” as Yahoo! calls them).

Reference 84

Social navigation may also help in knowing what information is reliable or trustworthy. In the physical realm, identities are relatively stable, reputations and brand names are built up over a long time, and centralized certification authorities such as the American Medical Association (AMA) act to ensure a reasonable level of reliability. In the on-line world, and on the Web, identities are fluid, reputations are more easily forged, and certification authorities are rare or nonexistent. Some work is beginning on new forms of reputation assignment and management but for now we mostly need to rely on our friends. In particular, we can get recommendations from our friends’ Web pages in much the same way we get recommendations from our friends. That is, if I have a friend who is an AMA member, then the links that he has made from his Web homepage to other medical information are much more likely to be reliable than links returned from a search engine or from a self-selecting mechanism. Of course, this depends on our knowing someone who is both an expert on a given topic and who has taken the time to select, organize and maintain a set of links. In most cases, there is no such person known to us. One reason is the amount of work necessary. In addition, when we become expert practitioners in an area, we often omit from our written accounts or spoken instructions — because we expect it to be understood — precisely the sort of information that would help newcomers.

*Who is an Expert?
The Maes car-shopping
story*

We also need not confine ourselves to traditional “experts.” Each of us who has found useful solutions to problems can be considered skilled practitioners in that we solve these problems; as a result, each of us has knowledge that could be fruitfully captured for later use. For example, during the research for this dissertation, my advisor bought a new car. At the start of this process, she did a significant amount of Web navigation. She visited car manufacturer sites, car dealer sites, car magazine sites, and so on. The goal of this process was not to find a particular item; in fact, at the end of the process she had more alternatives than she had at the start. In the terms of our earlier discussion, what she did was to explore the space of possibilities. At the end of this exploration, she had a body of knowledge: what sites she had visited, what useful things should have been connected but were not, and so on.

This knowledge did not per se make her an expert; however, it was a valuable experience. Given that she had such experience, the logic of social navigation indicates that it would be beneficial — in the sense of saving others’ time and in the sense of increasing reliability — to be able to ask Maes about her experiences. Of course, to be able to ask her, one needs to know that she has this knowledge. Plus, she needs to have some inkling that others may be interested in what she has done and particularly how she did it. Maes’ list of generated alternatives is unlikely to be of interest to any other person; however, the work she did in exploring the space could potentially be quite valuable. For example, which sites did she find to be the most useful; what is the best order for visiting these sites, and so on.

Her exploration by itself is not nearly so useful as her exploration combined with the navigation done by others. This communal information source is much more likely to be useful to future explorers than any one individual’s work, unless that one person is a true domain expert; and, even then, it is hard to convey expertise indirectly.

3.6 Footprints Tools

Since most of us are not experts, we have created tools to help any Web navigator take advantage of interaction history. These tools operate together, using data gathered from many users. As people navigate the Web, their interaction histories are recorded

and made available to others. The rest of this chapter describes the tools built during investigation for this dissertation. These tools are intended to serve three purposes:

- First, they should demonstrate that interaction history can usefully be made part of a system in the digital realm.
- Second, they should serve as experiments to help explore different aspects of the space of all possible interaction history systems, as described in Chapter 2.
- Third, they should aid users' information navigation tasks by enabling a community of people — even though distributed in space and time — to share guidance and experiences.

As a guiding metaphor for our navigation tools, we picked three kinds of artifacts that are commonly associated with navigation in the physical world: *maps*, *trails*, and *signposts*. Maps, as described above, let us show large-scale user experiences and, to use Lynch's terminology, help users identify districts and edges. Trails are equivalent to Lynch's paths, with documents serving as nodes. Signposts do not have a simple equivalent in Lynch's taxonomy; rather, they are associated with both paths (as guidance), nodes (as identifiers), and edges (as demarcations). Signposts are also often used as landmarks, though that frequently requires a physical distinctiveness and visibility that Footprints' signposts do not possess.

Metaphor for the Tools

Maps in our system are the highest-level view given of interaction history; they show all the transitions that users have taken to and from a document of interest and the documents connected to it, as explained below. Trails are a lower-level view, intended to convey the sense of "walking along" a trail where one can see the sequence used to reach a given point, as well as possible branching in the trail ahead that may lead to different end points. Signposts provide textual commentary on the trail parts — the nodes and links between the nodes. Users can express opinions, give directions, or explain why they went somewhere.

Clearly these are not the only possible tools that could have been built — other artifacts are used in navigation and could be implemented in a Footprints-style system. In addition, there are many possible implementations of any particular tool. As Hutchins shows, the form of the tool evolves to be reflective of the social practices within which it is embedded. Ship navigation charts are not just maps; they are communal artifacts tuned to a particular need and expertise of practice. Our tools will no doubt evolve as we gain experience with using interaction history in the digital realm. In fact, the tools discussed here are in their third generation, with features and functionality based in part on feedback from our users, beta testers, and usability test subjects. Feedback and formal test results are discussed in Chapter 5.

Reference 39

Footprints tools are designed to work on the World Wide Web, using standard Java and a small amount of Javascript. Thus our tools can be run on any Web-connected PC, Unix box or Mac that has a recent version of Java. Our tools work with Netscape Navigator and Microsoft Explorer 4.0 or later. No modification to the browser is necessary, in order to be as widely usable as possible.

3.7 Footprints on the Web

As noted in the preceding discussion, interaction history works best when the work of many people can be made available to any given user. This principle guided many of the decisions we made in implementing features for the tools — where a trade-off had to be made, we tried to decide in favor of keeping our software as widely available as possible.

The meaning of “browser”

Unfortunately for the following discussion, the word “browser” is overloaded. It can mean both the person doing the activity on the web, and the software that is being used. I will try to use phrases such as “browser software” to keep the meaning clear where necessary.

Footprints operates by means of a proxy server. It monitors the transitions users make between Web pages, and accumulates this data in a central database. When users turn on the Footprints tools, data relevant to their current pages is extracted and used to create visualizations to help the user. The following sections describe the basic data types used by the system, the rules for composing these types, the problems inherent in recognizing and validating them, and then covers the tools that operate over them.

The Web-navigation primitive

For web navigation, the primitive element of data is a transition:

$$A \rightarrow B \qquad \text{(EQ 1)}$$

where A and B are web pages visited by the user. Page A is visited first, then page B. A is called the “referrer” of B in that it refers to B by some means, as described below.

In the current set of browser tools commonly available to people, there are three ways to get from one page to another:

1. Users can click on a link in a page either directly or indirectly, for example via an imagemap that contains embedded URLs (Uniform Resource Locators).
2. Users can type in a URL — perhaps sent to them in an email message — or pick a page from their bookmarks, or from the history list.
3. Users might be taken to a new page without taking an explicit action. This can happen via server redirection (the remote web server sends requests for one page to another page) or by some form of time or active code in the page.

The only difference among these methods in our case is that the second method involves a link that the author(s) of the page had not included in the source of the page. This is important for two reasons. First, it reveals links that the user may think ought to be made between pages. These may be links that are obvious or interesting to readers but that were not included by the author for any number of reasons. This will be discussed in more detail below.

All transition methods create valid history

For the Footprints tools, we accept all three kinds of transitions as equally valid and do not distinguish among them. Our software reads log files, which capture all transitions. However, we do not track the use of the browser program’s “Next” and “Back” buttons. Although these buttons are often used by people to skip back and review past pages or set out on new directions after coming to a dead end, it is technically difficult to capture this usage. Our only means of doing so would have been to modify the user’s browser software, which we did not want to do, since it would limit the ability of our tools to be widely deployed. To this extent, we are using an incomplete picture of browsing history in our data; but this is inevitably the case.

URL equivalence

The question of what constitutes a document is a nontrivial one in the current environment. If URLs were more uniform, we could assume that a URL was the identifier for a document and if two URLs were different we could assume that they were different documents. Unfortunately, this is not the case. There are two kinds of variation in URLs:

- *incomplete (relative) URLs*. The URL “wex-grayscale.gif” appears in my home page. This URL is missing most of the components of a fully formed URL, but it is understood correctly because there is a declaration at the head of the file that gives the basic path to which this is relative. Unfortunately, many Web servers write relative URLs into their log files and we do not have the base path to work with. The proper solution to this is to form URLs into URIs, — the standard complete, location-independent Uniform Resource Indicator — before writing the URLs into files. Regrettably, this is not done by servers. To make URL comparisons easier, Footprints transforms URLs into URIs before making comparisons, but this does not solve all problems.
- *equivalent URLs*. There is usually more than one way to reach a document on the Web just as there is in a file system. These equivalent URLs are often formed because of links within the file systems that URLs trace. A more difficult form of equivalent URL comes about because machines can have different names. For example, my home page can be reached as

Reference 6

`http://wex.www.media.mit.edu/people/wex`

and also as

`http://lcs.media.mit.edu/people/wex`

Transformation into URIs will not solve this problem, since the URLs are already well-formed. To cope with this variation, we must consult the documents themselves. The contents of the document can be described by a unique hashing function; in our case we use the MD5 function.¹

Since retrieving the document and computing its MD5 is expensive, we do not want to do this unless we must. Thus the test for document equivalence is as follows:

1. If the URLs are the same, the documents are the same; or
2. if the URIs equivalent to the URLs are the same, the documents are the same; or
3. if the MD5 hash values of the documents retrieved from the URIs are the same, the documents are the same; otherwise
4. the documents are different.

Test for document equivalence

Unfortunately, things are not as simple as this. We also must deal with the irregularities of Web document authors. In particular, we found it common to receive malformed HTTP (HyperText Transfer Protocol) streams from remote servers. Commercial browser softwares spend significant effort dealing with these problems; as a research system, Footprints often found itself unable to deal with the peculiarities of bad data we received.

Contrary to what we expected, most of these bad data came from the commercial sites, not from the amateur sites that people traversed. Commercial sites compete to have the best effects and the slickest interfaces. The result is often extremely hard to parse or malformed in some untraceable way. This turned out to be particularly problematic in the user testing because our task called for users to visit a large number of commercial Web sites. Some test subjects’ data had to be discarded because their interaction with the commercial sites caused problems for our software.

See “The Task” on page 112.

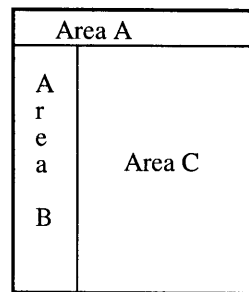
1. A hashing function turns a source text into a representative number called the hash value. The MD5 function uses a set of representative numbers which is large enough that two randomly selected documents are extremely unlikely to produce identical hash values.

What is a page?

Finally, there is a fundamental problem of what to consider a “page.” In an ideal world — and from the user’s point of view — the “page” is that thing you see on your screen as a unified whole. In fact, a page that appears unified on a screen is more often than not the result of many separate loads. My home page, for example, loads a separate image from the local disk, an image from a remote server, and a Java applet from still another server. Commercial sites often include advertisements from remote sites as well as many images.

For the most part, we can screen out nontextual media types by looking at the properties of the HTTP stream. However, it is perfectly possible for a “page” to consist of several HTML documents. Typically, these are loaded inside something like an HTML frame, which is diagrammed below in Figure 8.

FIGURE 8. Stylized representation of HTML “frames” page



In this picture, the user’s screen is apparently divided into three “panels,” each of which is coded as an HTML frame. In each frame a separate document is loaded. Each of these documents is itself an HTML page. The user, however, interacts with each part of the page separately. Layouts like this are often used to allow users to have a stable navigation area (B in this diagram) and a stable banner area (A in this diagram) while changing the data displayed in the main area (C). Each of these areas has its own URL and corresponds to a fetchable document.

Rather than trying to track all the subdocuments associated with a given “page,” Footprints makes the simplifying assumption that only the first textual thing loaded by the page (whatever the area or layout of the page) is important and associates all data with that element’s document. We can detect the order of loading simply based on the order in which the frame components are listed in the HTML page. This heuristic works in most of the cases we have encountered so far.

3.8 Enhancing the Data Structure

Given the primitive data type of the A→B transition, the question becomes how to enhance it. There are a number of different attributes that we could use. For example, attributes of the pages such as type of content, size of content, recency of use, recency of modification and so on could be recorded. Additionally, properties of the person could be recorded, such as user name, reason for going to the target page, level of web expertise, and so on. Finally, properties of the transition itself could be recorded, such as the times of accessing documents A and B and the method of getting from A to B.

The interaction history framework of Chapter 2 suggests guidelines for what kinds of information could be gathered. Our goal in designing the Footprints tools was to give users assistance in navigating — specifically, assistance in finding information they

wanted and assistance in putting found information into context with other pages in order to aid users' understanding of the pages.

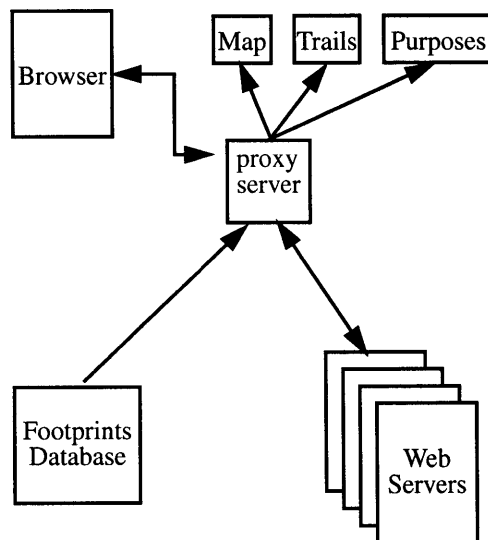
In addition, privacy was a major concern for us. Tracking what Web sites users visit is a potentially serious privacy invasion. This problem is compounded when — as in our case — the Web sites visited by one user might be made visible to other users. The problem is not simply that someone might know which porn sites you have been visiting; employers might be concerned if they found employees checking out the competition's Web site for possible new job opportunities. Liability issues might also arise if one company found out that employees of another had been visiting their Web site often in the months before the release of a competing product.

Concerns about privacy

Given these concerns, and the basic fact that research into implementation of privacy protection schemes is not part of the goal of this dissertation, we decided not to record or include any sense of the user's identity in the Footprints system. In addition, our Web site has an explicit privacy policy on it (<http://footprints.media.mit.edu/privacy.html>, reproduced in Appendix A). The policy says, in essence: "If you use our software, you agree to give us your data. In return, we agree that your data will never be identifiably associated with you." That is, anonymity is the only form of privacy protection we offer — all users' data are merged together and it is not possible, as will be discussed below, to extract the history information of any one user.

Users do not have to make one decision for all time, of course. Because the Footprints tools are controlled from a proxy server, users can turn off the monitoring capability with a single button push.² The same button turns the software back on. This was important for our usability, since alternative implementations we evaluated required the user to quit and restart the browser software. While this might have worked for our test subjects in the lab, it seemed unrealistic to ask users in real work or recreation situations to accept this limitation.

FIGURE 9. High-level Footprints Implementation Architecture



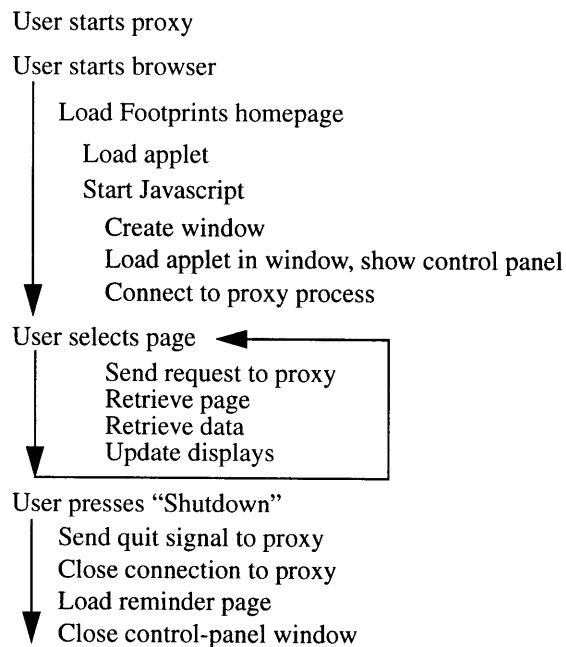
2. Technically, it is more than one button push, since the user must select an option from the browser software's preferences dialog and getting to that dialog require some amount of input before the button can be pushed.

*Components of
Implementation
Architecture*

Our implementation architecture, shown in Figure 9 above, has two major components: the database server, or back end, and the proxy server, or front end. Each of these components performs different functions, many of which are invisible to the user.

The user begins using the Footprints system by visiting our web page (*footprints.media.mit.edu*) and unloading from there a package (a Java JAR file) containing the proxy server and associated files. After this is unpacked, the user sets a home page within the browser software to be the special Footprints home page. This page loads the applet, which is part of the front end, and creates the control panel that allows users to manipulate the tools. The user then starts the proxy server, which runs as a Java program on the user's computer. Note that this architecture means that each user must run a separate copy of the proxy server. This can be a problem on multi-user machines, since the proxy listens on a dedicated port number; fortunately, this has not been a problem for our users so far.

FIGURE 10. Footprints Operation Sequence

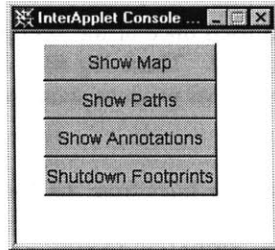


When the user starts up the browser software, the proxy server loads the Footprints home page. This page contains an applet that will communicate with the proxy server. Because of the limitations of the security model imposed by browser software the applet is limited in where it can communicate. Specifically, the applet needs to be able to communicate with the proxy server, as explained below. To accomplish this, the proxy server loads the home page from the local disk. With a locally loaded applet, browser software will allow the applet to communicate with the proxy server on the same machine.

The applet, when loaded, uses Javascript to create a window. This window contains a control panel, pictured in Figure 11 below. This window contains another applet to

handle the communication with the proxy server, because when the user moves off a page with an applet the applet stops running.³

FIGURE 11. Footprints Control Panel



The enduring applet is responsible for telling the browser software to load Web pages based on user actions and choices, as described below. The applet also gets input from the buttons in the control panel. These buttons control the various tools provided by the Footprints system and allow the user to shut down the proxy server.

If the user selects the shutdown button, the proxy server is sent a message to terminate itself, then the control panel window goes away, leaving an HTML page that reminds the user to either restart the proxy server or turn off the appropriate preference option to allow normal browsing without a proxy server.

The proxy server operates in most respects like a standard Web proxy server. That is, it accepts requests for pages, contacts remote servers, and uses HTTP to retrieve the requested pages. However, the Footprints proxy server has a number of enhancements specific to the Footprints tools:

- incoming HTTP streams can be parsed;
- an extensible number of interface windows can be created; most of the Footprints tools are implemented as these windows;
- a TCP/IP socket connection is maintained to the Footprints database server. This connection is used to retrieve data from the database as needed.

*Footprints proxy
enhancements*

HTTP stream parsing is used for two purposes. For streams that contain HTML text, markers can be inserted. For non-HTML streams, the parser needs to decide if it is worth querying the database. Since the network introduces unpredictable delays, considerable effort is invested in minimizing the number of requests sent to the database. We also optimize the traffic, as explained below. Since the database only stores documents that we can manipulate in some way — HTML text, raw text, and so on — images, opaque media objects such as sounds and movies, and executable or other binary formats will never be found there. The front end eliminates queries for objects that will not be found.

The interface windows are created and removed as the user requests tools be turned on and off. As explained below, some of the tools can create other windows in response to user actions.

*See “Comments” on
page 81.*

3. There are various programming tricks that allow this condition to be avoided, but they do not all work in various versions of various browsers. Since we wanted to provide the user with a control panel anyway, it saved significant engineering work to combine the two.

The connection to the database is used to retrieve data to show in the tools, if appropriate. The context for data is, obviously, the Web page about to be viewed in the user's browser. Data relevant to that page may include maps, paths, and comment information.

The database connection is also used once at the beginning of the user's session to transfer the log files to the back end. These log files are of two kinds: access and instrumentation. The proxy server does produce standard error logs, but these are not routinely sent to the back end.

Figure 12 and Figure 13 show sample lines from the two log files. The access log is in the standard referrer log format. This is identical to the format produced by most popular servers including Apache, CERN, and Microsoft's IIS.

FIGURE 12. Access log line

```
1           2           3           4  
18.85.16.11 - - [06/Aug/1998:13:06:55 +00] "GET http://www.media.mit.edu/MediaLab/People.html HTTP/1.0" 200 23988 http://www.media.mit.edu/  
5           6           7           8
```

1. IP address of the requesting host machine;
2. date, time and timezone of the request (surrounded by brackets);
3. HTTP method of the request (after open quote);
4. URL of the requested page;
5. protocol used in transferring the requested page (up to closing quote);
6. result of the request (200 indicates success);
7. size of the returned data (in bytes);
8. referrer page URL.

FIGURE 13. Instrument log line

```
1           2           3           4  
PathMap [04/Sep/1998:13:52:58 +00] http://www.media.mit.edu/unhighlighted
```

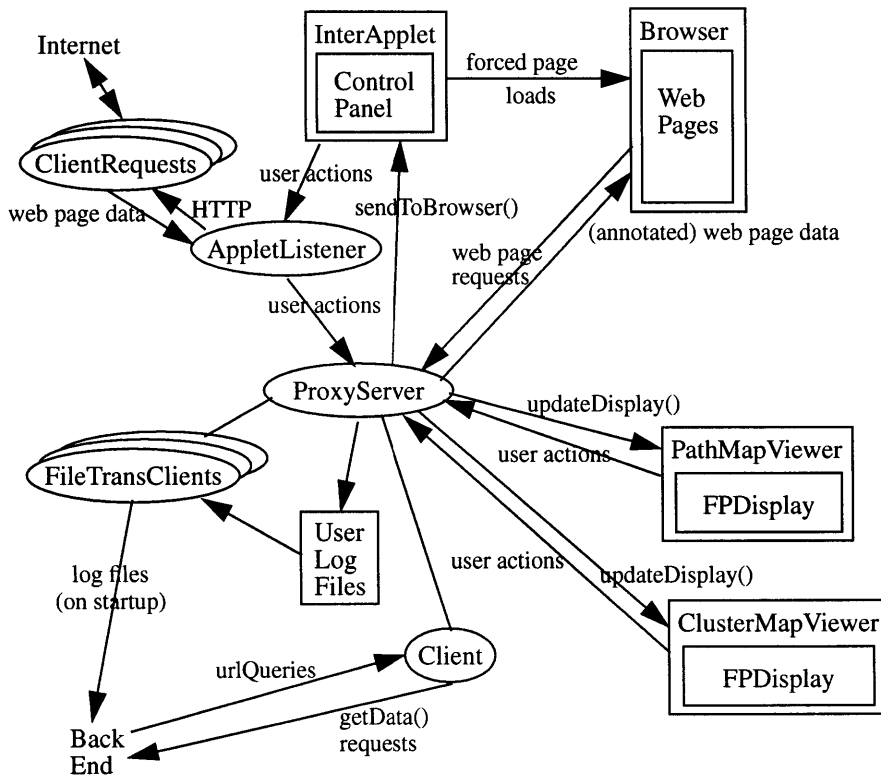
1. name of the tool with which the user is interacting
2. date, time and timezone of the interaction (surrounded by brackets)
3. URL of the document on which the interaction was done
4. kind of interaction (highlight, unhighlight, select)
5. [not shown in this example] records whether paths are present for a requested URL

The instrument log is a format we designed to keep track of how our tools were being used. Clearly a production-value form of these tools would not need such a log, but one of our goals has been to understand how our tools are used both in experimental and "field" conditions. Therefore, the instrument logs keep track of when each tool is used, when it is turned on and off, when it is on but has no data, and when the browser uses a Footprints tool to select a page. This includes viewing or adding comments, going to pages by using the path or map tools, or following an annotated link in a Web page.

Figure 14 shows the front end in more detail, including the data flows between components of the system. Things that are displayed to the user are shown as double rect-

angles with the name of the window at the top and the name of the displayed element in the box. Thus, the “Browser” window shows Web pages. Arrows show the direction of data that is transferred between components and the lines are labeled with the kind of data that is transferred.

FIGURE 14. Detailed Architecture of Footprints front end⁴



The front end is modularized and uses Java’s threads to enable multiple actions to happen simultaneously. In particular ClientRequests — which retrieve data from remote Web servers — run in parallel. Communication with the back end can also be done in parallel, but we are using a simple database for storage and retrieval that can only handle one request at a time, so all requests to the database must be serialized.

Communication with the database is via Java streams, a method of serializing memory objects for transmission (over a network or to disk). The streams transfer objects via a very direct algorithm, so efficiency is not their best feature. Consequently, the Client section of the front end uses several techniques to minimize the amount of data that actually gets transported.

The most obvious technique has already been discussed: don’t ask for data you can’t get or won’t use. The Client will not request data for objects of kinds not in the database, and the Client knows which tools the user has selected for use. If a tool is not in use, data for that tool is not sent. In the case of the map, this simple principle repre-

Traffic minimization techniques

4. From an original diagram used by Felix Klock to teach newcomers to the project about the system’s architecture.

sents a significant saving since transferring some of the larger maps can take a minute or two to send from the back end.

Our next optimization is to compress (using the *gzip* standard) the data stream itself. This optimization is somewhat heuristic, since we do not check to see if the compression actually decreases the size of the data before sending it. In general, we discovered by informal testing, *gzip* will only increase the size of the stream if the amount of data to be sent is already small, generally under 10K bytes. At that size, the expansion due to the *gzip* algorithm does not increase the total data size significantly. At the other end of the scale — in our best cases — *gzip* gives us approximately a factor of 10 improvement in the size of the data to be transmitted (1.5MB reduced to under 200K). Thus we felt it was worthwhile to leave the stream compressed.

Our final optimization is to employ caching. Since the most costly items to transfer are the map clusters, the Client maintains a cache of the ten most recently used clusters. The IDs of these clusters are sent to the back end with the `getData()` request. This is necessary because the communication between the front and back ends is stateless. That is, the back end does not distinguish one Client from another and has no way to know which clusters are in the Client's cache.

The back end compares the clusters that would be sent in response to a query with the cache list and only sends new clusters. This is an enormous saving, since the most common move for users to make is to pick a node in the neighborhood of the current page. They often pick this node from the map shown or from a link on the page. It is rare for users to jump to completely new and unrelated sections of the Web. As long as the moves are local, the clusters are likely to be in the cache. Thus, the percentage of cache "hits" is high, and much less data has to be transmitted. In our informal testing, we found that this caching improved network performance by another factor of 10 and that this speed-up applied roughly equally to small and large maps.

Footprints Back End

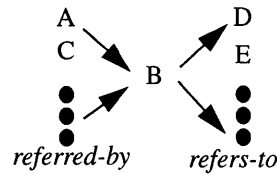
The final component of the Footprints architecture is the back end. This primarily consists of a database that sends history information to proxy servers as they request it. We have used a commercial Java database called JIOMA (www.w3apps.com) under an educational license.

Our instance of JIOMA runs on a PC in our lab. Data are fed into the database in two ways. History data, in the form of log files, are added to the database on a daily basis via an overnight batch job. Since history is a slow process, it does not seem necessary to have the database updating continuously. However, comments are directly entered by users — a form of active history — and users expect a response to commands and data they enter. Therefore, the proxy server transfers comment text to the database as soon as it is okayed by the user. These data are available to any user viewing the object to which the comment is attached, as soon as the database registers the new comment.

The Etrans

The database contains Java classes that represent Web documents, links among documents, and comment information. The basic transition structure ($A \rightarrow B$) is built up into a number of other more complex structures that are used by the interface tools. The first of these is the extended transition (or *etran*s). An *etran*s is a document and all the documents that have been used to reach it (called the *referred-by* list) and all the documents that have been reached from it (called the *refers-to* list).

FIGURE 15. Etrans



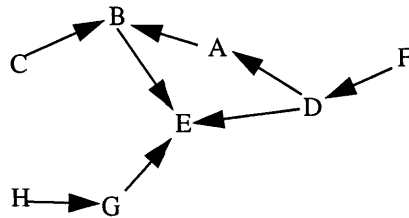
The refers-to and referred-by relationships are implicative, but unidirectional. That is, if document A is on B's referred-by list, then B must be on A's refers-to list. Documents may refer to each other in that they have reciprocal links in their text, but unless users specifically traverse these links, they are not entered into the database. If they are entered in, then the etrans will include the documents reciprocally in both relationships.

Extended transitions are further merged into clusters by a breadth-wise expansion from a starting etrans. A cluster is a collection of nodes that have some kind of associative relationship to each other. Each document in the refers-to and referred-by lists is added to the cluster, with appropriate links. Then the refer lists of these new nodes are collected and processed. As each page is examined, if it is already in the cluster, then its refers links are simply added to the existing node. If new documents are encountered, nodes are created for them. This process is repeated until the cluster is extended as far as possible. Then, if there are any etrans still unprocessed, one of them is selected and a new cluster is created.

The Cluster

The result is a series of clusters like the one pictured below; essentially a directed lattice of nodes. Each document is guaranteed to appear at most once in the cluster set.

FIGURE 16. Cluster



The final representation maintained by the system is the trail. Trails are linear sequences of pages visited by users that are assumed to have a common theme or topic or to be related to something particular. Normally, these are just the sequences of pages recorded in the access log files. However, this is not as simple as it seems at first. There are two problems that must be solved in building trails:

Trails in Footprints

- when does a trail end?
- when are two trails the same?

The first question would not be difficult if the user started and stopped the browser software with each change of interest. Of course, this is not how we use the software; we start and stop "surfing" more or less at will. We leave the program running for hours or days. We change our interests as we read. Since a trail should represent a coherent series of documents, we need some way to estimate when a trail ends, given the input of the proxy log files.

Trail segmentation heuristics

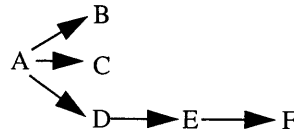
To do this, we use four heuristics, each, to some degree, based on intuition and experience. However, they seem to produce acceptable results and are easy to refine. It would be an interesting experiment to empirically test and verify the relative strengths of these or other heuristics.

Path Length

The first heuristic is length. Paths under a certain length are uninteresting and may even represent false starts that could confuse later users. Currently our length heuristic is set at 2; that is, only paths containing three or more pages are included in the data shown to users. Shorter paths are kept in the database, since they may later be merged with longer paths. However, we filter them out of the data before maps and trails are built. As the database grows, a simple way to help the tools scale would be to increase the minimum required path length.

Filtering out short paths eliminates a number of “false starts” such as illustrated below.

FIGURE 17. Typical False Start Pattern



In this diagram, A is a page with many potential outgoing links to be followed — for example, a results page returned by a Web search engine. In this behavior (that we call the “in and out” experience) the user has selected page B as a likely candidate but, after examining it, has decided that it is unsatisfying. He then uses the browser software’s “Back” button to return to the search page — remember that we cannot directly track the use of this button; instead we infer its use from patterns such as this in the data. Clicking on page C he finds it similarly disappointing and returns to the search page once more. Finally, page D holds some interest and it leads to pages E and F.

Our assumption, then, is that the trails A→B and A→C are not useful, but the trail A→D→E→F is useful. Of course, it is possible to describe scenarios in which pages B and C were important; they may have explained concepts that helped our user understand why page D was important. Indeed, we believe this is important behavior and show some interesting applications of it below. However, these scenarios are small minority cases compared to the hundreds or thousands of times when links are fruitlessly followed. Footprints is not aimed at a perfect solution; instead it is intended to be helpful, and that means we can accept missing some specialized cases.

Reference integrity

The second heuristic is reference integrity. Lines in a server log file may appear sequential but in fact represent interleaved actions. This can easily happen if the user opens multiple windows in the browser software. Although it seems redundant, we must check that the referrer field of each link matches the target field of the preceding link. That is, given the log lines A→ B, and B'→C we must ensure that B and B' are in fact the same document. This can be complicated by the many forms of URLs used by different documents, as noted above.

See “URL equivalence” on page 54.

Finally, users’ browsing can be discontinuous. They may use the back button several times in a row or select a document from their browser software’s history list. It is not uncommon to see entries in the server log that have the form A→B followed immedi-

ately by C→D particularly when C is a document visited by the browser software earlier in the session. In this case, we make a break in the trails — ending one at B and beginning another at C.

The third heuristic is session boundaries. The proxy server transfers the accumulated access log file to the back end each time it starts up. Trails are computed within a session file initially. This division may be arbitrary, as users may end a session for any number of reasons, including bugs in the software, a need to reboot their PC, or the end of a business day. In fact, many browser tools have an option that allows them to start up with the last page visited by the user as the default first page seen. Fortunately, we can account for these cases when trails are merged, as described below.

The final constraint is the most interesting.⁵ The constraint we use is that the time between page loads must not be too long. We call this interload interval *dwelt time* since it reflects the time that a user dwells on a particular page. The basic assumption of this constraint is that users will pause in their Web browsing between their work tasks.

Presuming an information foraging model such as the one presented above, we might expect that users' browsing would pause when they have achieved a successful exploitation, either deliberately or by accident, or when distracted by other tasks, or when they run out of energy, patience or time.

A research study by Morita and Shinoda showed a correlation between dwell time and user judgments of a page's utility. Specifically, the length of time spent on a page was a good first-order predictor of the usefulness perceived by the user. Of course, there are limits to this sort of prediction, particularly as we move away from laboratory conditions and into general office or home use of a Web browser. For example, if you stop browsing to answer email or to go get a cup of coffee then your dwell time will increase but that is not particularly relevant to the usefulness of the page.

More importantly for our present problem, we need to know if dwell time is correlated with connectedness. That is, does dwell time give any indication of when an apparently linear sequence of steps should be broken into two trails? Our assumption is based on the idea that when people have reached a goal or natural break point in their cognitive processing they are more likely to pause in their Web browsing. We take any dwell time of greater than two minutes to be an indication of a pause in activity and our heuristic causes a trail break at that point.

So, given the sequence A→B→C→D→E→F in a log file, if the dwell time on page C was greater than two minutes we would assume that this sequence represented two distinct trails: A→B→C and D→E→F. This assumption can fail in a number of cases, of course. Interruptions in activities may not be under the user's control, resulting in pauses that do not correspond to natural breaks in the task. However, since we combine data from many users when making trails, it is unlikely that all users will suffer involuntary interruptions at precisely the same point in a sequence of document visits. Thus any later user can correct a break erroneously created by this heuristic.

The use of a two-minute cutoff point for dwell time is also somewhat arbitrary. We examined log files for several hundred thousand page accesses over 8 months on the Media Lab Web server to arrive at this heuristic. Lian's work (in press) showed an

Session Boundary

Dwell Time

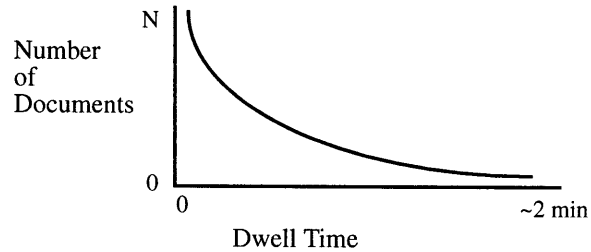
See "Alternative Model of Navigation" on page 46.

Reference 50

5. In fact, Alex Lian — one of the undergraduates involved with writing the proxy software for Footprints — went on to do his Master of Engineering thesis on this issue.

asymptotic distribution of dwell times with an approximately log-linear curve fit for most pages, similar to Figure 18.

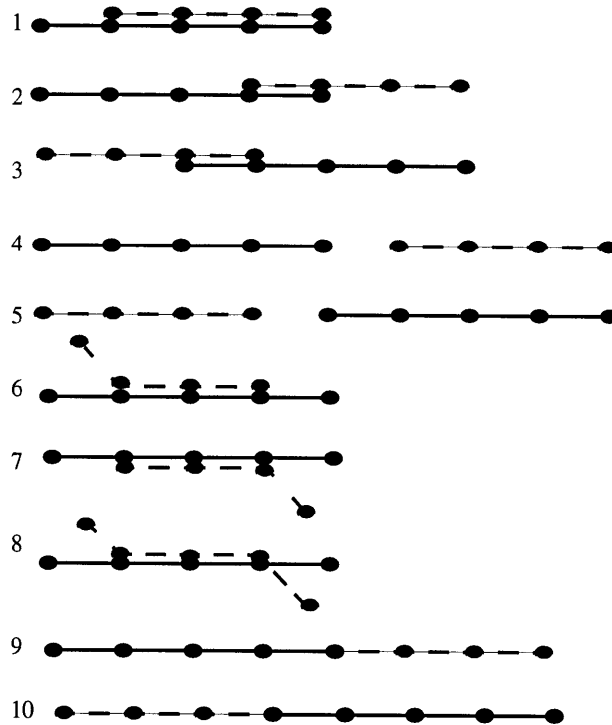
FIGURE 18. Distribution of Dwell Times



Fewer than 2% of dwell times exceeded the 2-minute threshold. The general shape of the curve holds true under both general Web browsing, as measured by the Media Lab's server logs, and task-specific Web browsing, as measured by graphing the logs generated by our user-test subjects.

Each of the preceding heuristics gives an indication of when an apparently linear sequence of page accesses should be considered two or more distinct sequences. Our next problem is the converse: When do we combine two sequences and consider them to be one trail? We identified ten possible ways two sequences might be related to each other. These are diagrammed in Figure 19.

FIGURE 19. Possible Methods of Combining Two Sequences



In the diagram the solid line represents one sequence, the dashed line another. Dots within the line represent distinct nodes (documents) in the sequence. As a heuristic guide for determining sameness we use a loose analogy with the U.S. Interstate High-

way system. That is, two sequences are the same if and only if one sequence completely contains the other (case 1) or if they are contiguous for at least one segment — two nodes and one link (cases 2 and 3). The negative cases are largely uncontroversial. Discontinuity (cases 4 and 5) is clearly different. The overlap cases (cases 6, 7 and 8) are different both from inspection and from our analogy — highways may overlap for some of their distance and yet not be the same.

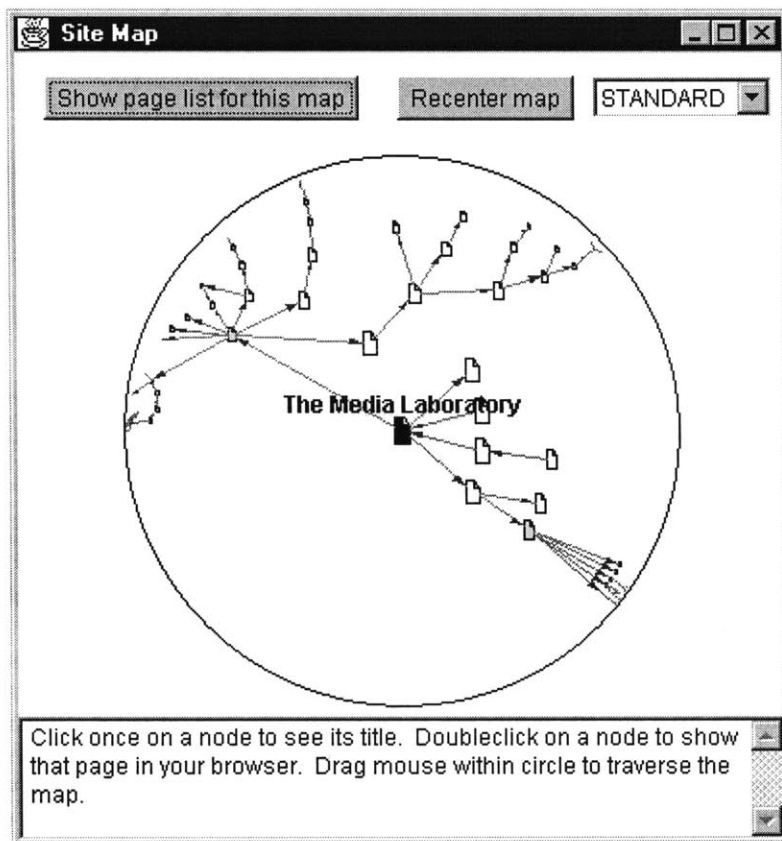
The abutment cases (cases 9 and 10) are the hardest to judge. These cases share only one node — the end of one is the start of the other. By our analogy we do not combine them: one highway ends where another begins. The argument from our dwell time constraint above is less clear. Currently we maintain the dwell time constraint, but rather than using a single dwell time, we check the median dwell time. If the median is over two minutes, then we do not connect the two paths, otherwise we do.

Other cases can be decomposed into variants on these basics. For example, a trail might be constructed out of cases number 6 and 7 that involved leaving the path for a time, then returning. This would not be the same. In the general case, any variant that includes a non-match component is itself judged to be non-matching.

The etrans, cluster and trail representations are used by the Footprints interface tools. The first such tool is the *map viewer*, which gives users the high-level overview of the history information. In the current version, we give browsers the option to select from three different kinds of maps.

Basic Footprints Tools

FIGURE 20. Footprints Standard Site Map



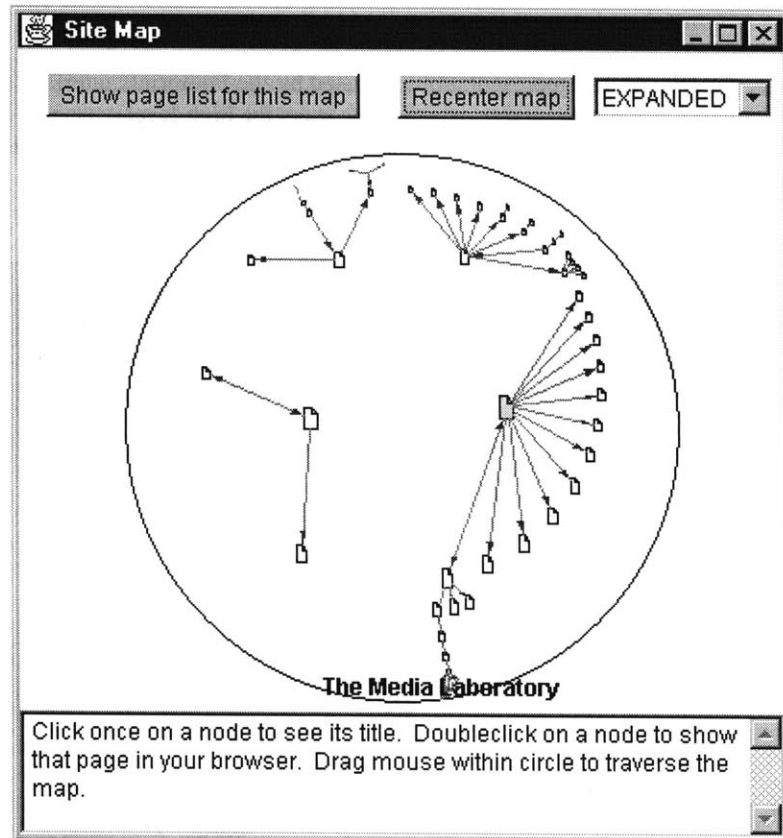
3.9 Maps

References 80, 81, 82

The standard map, selected by default, is what we have called in the past a “site map.” That is, it is a representation of the history of use of a given Web site, relative to a target document. This is pictured in Figure 20. When a user visits a Web page, the URL of that page is passed to the back end, which then extracts from the database all clusters containing any documents on that site. So, if someone visits my home page — *http://wex.www.media.mit.edu/people/wex* — any clusters containing any documents with URLs from the site *wex.www.media.mit.edu* would be included in the map.

The second map form is the “extended site” (pictured in Figure 21). This form recognizes that site names are often several levels deep and allows a wider exploration. For an extended map, the starting URL is left-truncated to give the parent site name. In our example, that is *www.media.mit.edu*. Then all clusters containing any document from that site are extracted and put into the map.

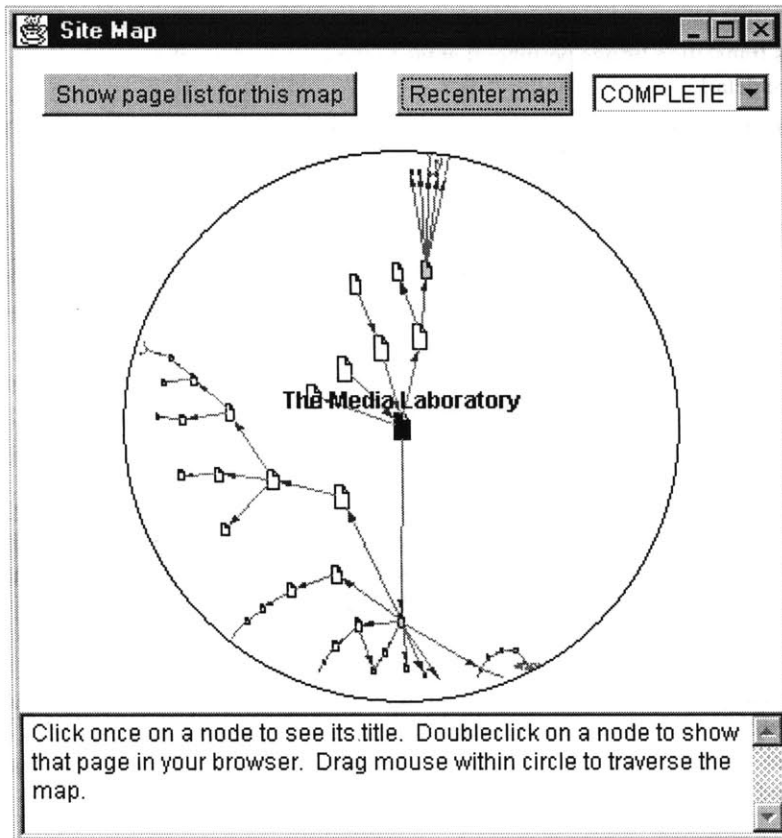
FIGURE 21. Footprints Extended Site Map



Finally, we allow users to see a “complete” map (Figure 22). This map is complete in that it involves a transitive closure of history information relative to the source document. This is accomplished by finding all sites for all documents in the cluster containing the target document. This list is then iterated over and all clusters containing any document from any site mentioned in the original list are retrieved. This multi-site map is often quite large, but it provides all the context necessary for understanding how people are using a particular piece of information. As discussed below, this kind of view is particularly useful for information designers, even if it is less useful for information navigators. Obviously there can be many different ways to pick

which clusters to show; these three varieties of map were selected based on feature requests from users.

FIGURE 22. Footprints Complete Map



The map tool has a number of features to help users orient themselves. The basic view is a “fish-eye” that provides focus and context. The focus is the area in the center of the map, which is shown in high detail and larger size. The context — where the focus area fits into the map as a whole — is shown around the periphery of the focus area. Our map viewer is based on work done by John Lamping and his colleagues at Xerox PARC, but has a number of Footprints-specific enhancements.

Reference 25

Reference 43

The layout algorithm is a simple breadth-first traversal of the cluster structures. We do not optimize the layout (e.g. by rearranging nodes to reduce line crossings). This leads to faster layout and redisplay, but reduces usability somewhat. The basic notion is to lay the graph of the map out on a hyperbolic sphere, then project that sphere down to the flat plane of the display screen. The result is some nonlinear distortions of the layout such that lines which might be thought to be straight appear to curve as they approach the “horizon” of the display.

The layout works well for the kinds of clusters we are using. The only problem we found was for some experienced users who are used to displays in which the distance between two elements is semantically meaningful. Two nodes being farther away means something different from being close together. In our display, of course, this relationship does not hold — nodes near the center of the map appear farther away

from each other so more information can be shown, while nodes around the periphery appear close together.

The map can be manipulated by users in two ways:

- the left mouse button drags any portion of the map to any area inside the circle. This translation brings peripheral areas to the center where they can be seen in more detail.
- the right mouse button spins the map around its current center. This rotation keeps the focal area in place, but can make reading the map easier.

Interaction with the map nodes also takes two forms:

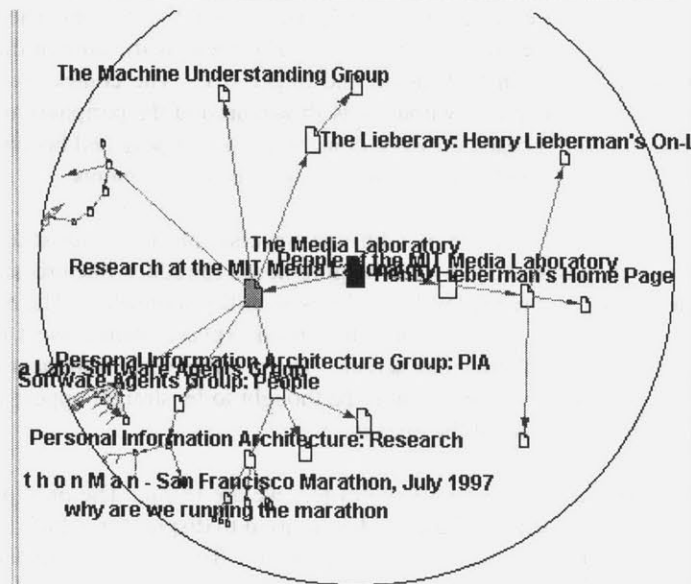
- users can single-click on any node to bring up the title of the document corresponding to that node. The title is displayed centered over the node. Single-clicking on a node that is displaying a title hides that title.
- users can double-click on any node to bring that node up in the corresponding browser window. The selected node is colored black in the map for ease of identification.

In addition to the black color used for the current selection, we color code some of the nodes in the map to show the “hot” documents, in the sense of the documents that get visited the most. For hotness we use red, so nodes in the map are shaded from white for least used, up through pink, to red for the most used.

Elements appear in the map if they are part of any relevant cluster, as defined by the rules of the map type. All nodes that are relevant are included in the map, regardless of how many users have visited them. This means that there are often nodes on the map that do not appear on trails, since the rules for trails are more strict, as was explained earlier.

See “Trail segmentation heuristics” on page 64.

FIGURE 23. Cluttered Map



If users get “lost” in the map — that is, find themselves somewhere on the display and do not know how to find what they are looking for, there is a “Recenter Map” but-

ton to help them. This button redraws the map, centered around the node in which the user most recently demonstrated interest, by either single- or doubleclicking. If no node is relevant, the map recenters around its own origin.

This facility maintains a history so users can backtrack simply by unclicking on nodes and recentring. That is, if I have clicked on nodes A, B, and then C, I can retrace my steps by clicking on “Recenter” (which centers on C), clicking on node C to unhighlight it, then hitting “Recenter” again to cause the map to center on node B, and so on.

Showing more than a few titles at once causes the map to become extremely cluttered, as in Figure 23, so we provide a pop-up window with an alphabetized list of the titles of all documents in the map. This window is controlled (popped up and taken down) via a button in the Map window called “Show page list for this map.”

FIGURE 24. Footprints Map Titles Window

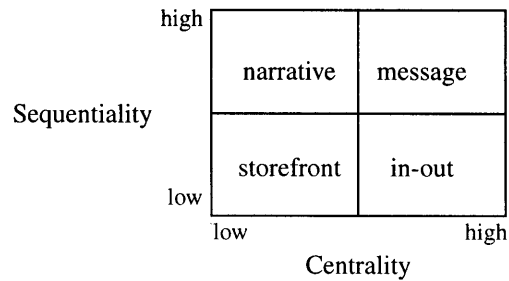


Maps represent the highest-level overviews of history information in our system. As a result, it is possible to see patterns in this representation of the data that are hard to see otherwise. These patterns represent kinds of user experiences; they form the basic units of navigation in our system, much as nodes, edges, trails, districts, and landmarks form the basic units of Lynch’s system.

History Patterns

We distinguish four basic units: *narrative*, *storefront*, *in-out*, and *central message*. We organize them along the dimensions of *sequentiality* and *centrality*, as show in Figure 25. These units are stereotypes, which are reproduced in variant forms in actual use, but they help us understand what is happening.

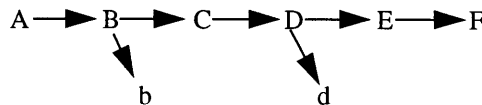
FIGURE 25. Dimensions of Navigation Experiences



Sequentiality refers to the order in which nodes are experienced. An arrangement with high sequentiality is likely to have a low branching factor and is likely to be longer. Centrality refers to the tendency to lead towards a node or group of nodes that function as the center of the experience. Centers are returned to often and contain items of high importance. Sequentiality leads to structures like Lynch's edges and trails; centrality is important for nodes and districts, though landmarks may also function as centers.

A *narrative experience* is something like being told a story. A typical narrative experience is shown in Figure 26.

FIGURE 26. Narrative Experience Example

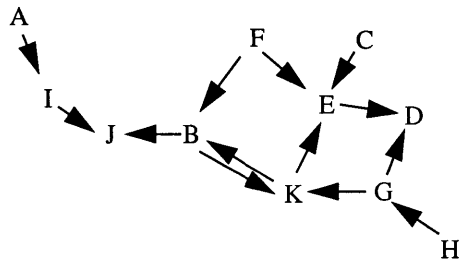


It depicts a linear sequence of nodes — tell me something, tell me something more — with minor branches off the main “story line” to get more detail or explanation. In Web spaces, these diversions are usually pictures or diagrams too large to be included in the main flow of text. A narrative experience is highly sequential, as most storytelling remains linear even in hypermedia environments. It is also not very central; the story may have a central point or narrative climax, but from the outside this part of the trail is unlikely to look any different.

A *storefront experience* is something like walking into a grocery store — shelves of merchandise with vague categorization and no obvious ordering among the items. This lack of order precludes any hope of sequence. There may be a storefront (or *portal*, in current popular Web jargon) but this entry point is rarely returned to by users once they have passed it. The information may be structured — as a catalog, as an archive, etc. — but models of how the structure can be navigated are lacking. These models must come from the user, and capturing the users' models is one of the most valuable purposes for interaction history.

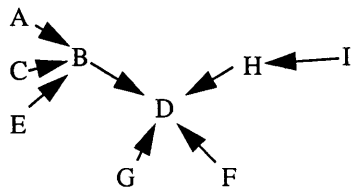
Storefront experiences are often quite confusing for new users, who find themselves encountering elements “out of order” and who may quickly become lost or discouraged. Conversely, experienced users often have less trouble with this kind of navigation, trusting in their familiarity with the space to make the irregularities feel proximate.

FIGURE 27. Storefront Experience Example



A *message experience* is one that has both high sequentiality and high centrality. Usually there is a central node or cluster of nodes around which the experience is focussed. A message experience may be pedagogical with a central theme or lesson the teacher repeats several times to help students remember it. Or it may be a commercial message where the external elements are product presentations and the central message is “give us your credit card number.” Sequence is important in a message experience because the message — the lesson, the sales pitch, etc. — will not be heard or understood properly if it is delivered out of context.

FIGURE 28. Message Experience Example



Finally, the *in-and-out experience* is one that is all too common the Web. Here the message pattern is reversed and instead of coming to a central node, browsers begin at such a node. From this node, they make a number of forays, usually in an attempt to find something that satisfies their immediate need. This back-and-forth behavior is usually caused by pages that have — or are nothing but — a set of links to other pages. These links may be more or less well described; unfortunately, they are often just a few words. Although, as Ellen Spertus showed, there is potentially a good deal of information that can be extracted from the link text, that information is conventionally not available to or used by ordinary Web browsers. Therefore, users must click on the link and visit the target Web page to determine if the target pages satisfies their need.

Reference 74

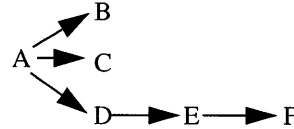
When it does not, as often happens, they move back to the first page and make another selection. George Furnas discusses this problem as one of knowing the “to set” for any given link. That is, even if the link text is descriptive and the target page is well-constructed, there is in general no good way to know what set of things might be reachable by following any given link.

Reference 26

The in-and-out experience is highly central because the page containing the list of links serves as a center point. However, the sequentiality of the experience is low because there is usually no obvious order to the links. There is, of course, the order in which they appear on the page, and some enhancements may be offered on the page to emphasize this ranking. For example, a search engine may rank its results in terms of presumed relevance to the user’s query and put the ranking numbers next to the links.

However, there is nothing to prevent the user from selecting suggestions in random order, say 2-5-3, and so on.

FIGURE 29. In-and-Out Experience Example



As with Lynch’s navigation elements, the experience elements described here can be combined in several different ways. For example, a Web site designer might create a narrative style experience that leads into a storefront. This way a naive or new user can get a gentle (albeit constrained) introduction to the material, while an experienced user can simply jump in, trusting her native navigation skills and past experiences to help her find her way around. Multiple narratives can also lead to a central message; this is a common pedagogical tool. Remember that these examples are stereotypes intended to illustrate the extreme positions, not a taxonomy of the only possible arrangements. Our data most often show hybrids of these experience types rather than the “pure” forms. We are using idealized types to illustrate a space of analysis. Designers may want to deliberately create hybrid experiences by changing the sequentiality and centrality of their information designs. For example, sequentiality may be enhanced by taking one long page and breaking it into sections, each of which is presented on a separate page. Centrality may be relaxed by taking a central message from one page and repeating it at several locations in the information structure.

It is also not our intent to assign value judgments to these kinds of experiences. Knowing whether any given experience is good or bad requires knowing both something about the content of the pages and the intention of the designer. We will discuss this in more detail below.

Finally, we want to emphasize that this experience-based approach gives us a level of analysis that exists above, but complementary to, the usual quantitative (hit count) methods of analysis. As noted in the previous chapter, interaction history is about processes and the artifacts supporting them.

3.10 Paths

Dropping “down” a conceptual level from maps, Footprints also offers a Paths (also called trails) tool, shown in Figure 30. Where maps are oriented toward a site, paths are oriented toward cross-site travel. Maps give a unique representation to each document; in paths, documents are represented at their proper places in the sequence. Thus a document will appear at least once in each trail, and may appear multiple times if the trail contains loops.

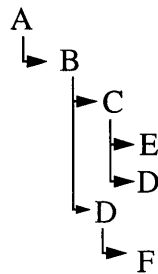
Since the number of trails in the database is quite large, only trails that contain the current document are shown in the viewer. As the document in the browser changes, the set of trails displayed is updated by the system. If the user is following a trail — either by chance in the Web pages, or deliberately in using the Path tool — the current path will stay in the window. If the user is “blazing” new trails then the set of trails displayed may change with each step. However, just as with maps, the most common thing for people to do is to move locally rather than jumping to distant points in the Web. The result is that much of the context is maintained from page to page.

To reduce clutter in the path view, we tried a number of different approaches. In the present version, we use an approach that emphasizes the metaphor of proceeding along a trail that may branch or fork and go in different directions. Our goal is to emphasize alternatives that users can pick to follow.

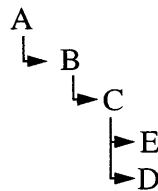
The visual implementation of this idea combines trails starting from the same points. This will, we expect, seem familiar to users who have had experience with navigating hierarchical file structures, such as those commonly used on PCs running Microsoft's Windows desktop systems or Macintosh computers running Apple's Finder desktop. One of the key differences from our path viewer, though, is that we do not retain on-screen trails not taken. As the user navigates down a path, new alternatives are shown "in front" of him, while choices not taken are removed. Of course, the reverse is also true: if the user chooses to navigate up a path, then alternatives that branch off earlier in the sequence will be revealed. For example, imagine that the following sets of paths are in the database.

A→B→C→D
 A→B→C→E
 A→B→D→F

Then if the user was looking at page B, the representation would be:



The two representations of 'D' are not collapsed since they are arrived at by different paths through the space. The multiple representations of A and B are collapsed, however, since their positions in the path space are identical. If the user now selects page C, the representation would change to:



The A→B→D→F path is removed from the display because it is no longer relevant. Other paths containing page C (that do not also contain page B) would be shown if they were in the database. If the user backtracks, the previous path is redisplayed.

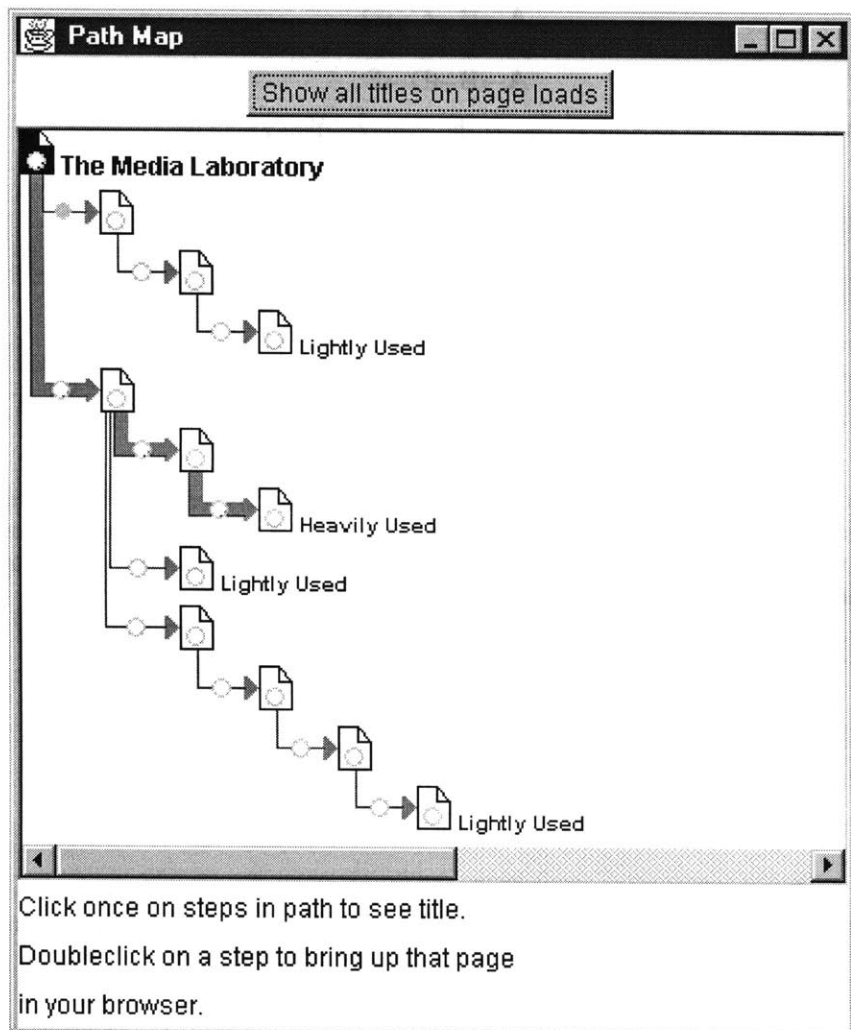
The path viewer is shown in Figure 30. As many similarities as possible with the style and functioning of the map viewer are maintained. Thus, the nodes are documents, and the links are transitions that people have made between them. Unlike the map, no bidirectional links are made in the path view. Traversals that go in both directions between two nodes represent either two separate trails or two separate steps on one trail.

Nodes in the path view respond to single- and doubleclicking: single clicks show or hide the titles of the nodes; double clicks select the node and bring up the corresponding document in the associated Web browser.

The path and map views are coordinated in the sense that selection and highlighting in one view causes the same document to be selected or highlighted in both views. This includes highlighting all occurrences of a given document in the path view simultaneously, as shown in Figure 31.

The path view usually contains far fewer nodes than the map, so we lay it out in a way that allows titles to be seen without overlapping each other or the nodes on the display. Unlike the map display, nodes in the path view are in a fixed layout. Users can scroll the view vertically and horizontally to see all the trails and the title text.

FIGURE 30. Footprints Paths Viewer

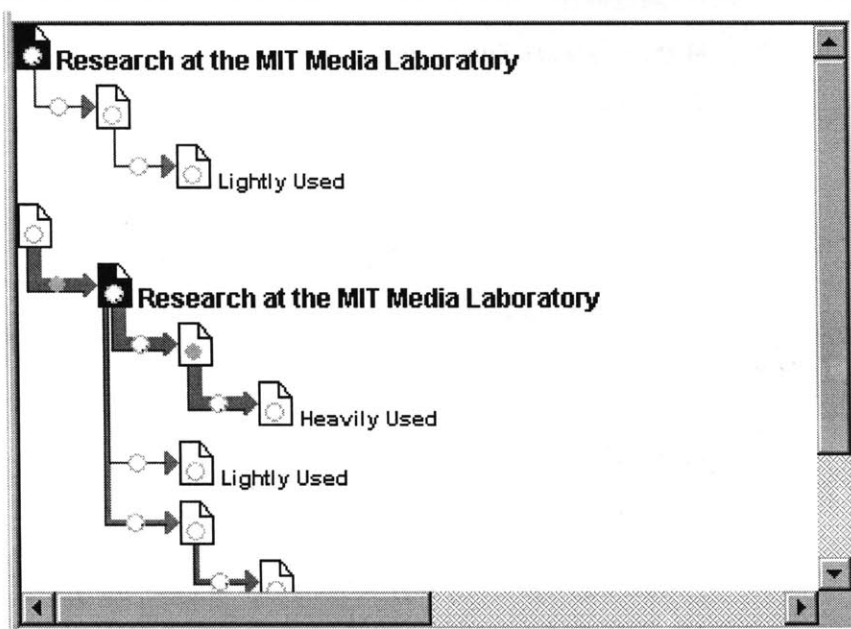


The path view also encodes degree of use. However, rather than showing the “hot” documents, as the map does, the path view shows the relative amount of traffic on a given path. The lines representing the links between documents are thickened as the trail gets heavier use. We use four levels of thickness to represent light, average,

heavy, and very heavy use. The level of use is also written as a text string next to the terminal node of each path. These levels — like the hotness levels of the map — are scaled relative to the entire database. This prevents all trails from trending toward the high end as more and more data are added incrementally.

Thickness information is propagated upward through the merged portions of the display, but not in a linear fashion. Each link shows the maximum use level of any of the trails that make it up. We tried a number of different algorithms for showing merged levels of traffic and all of them suffered from an inability to scale. That is, as the number of trails in the database increased, the display became increasingly cluttered and consequently less readable. Finally, we settled for simply using the maximum value, even though this distorts the mapping of line thickness to trail segment use. Our users do not seem to have a problem with this.⁶

FIGURE 31. Multi-highlighting in the Path Viewer



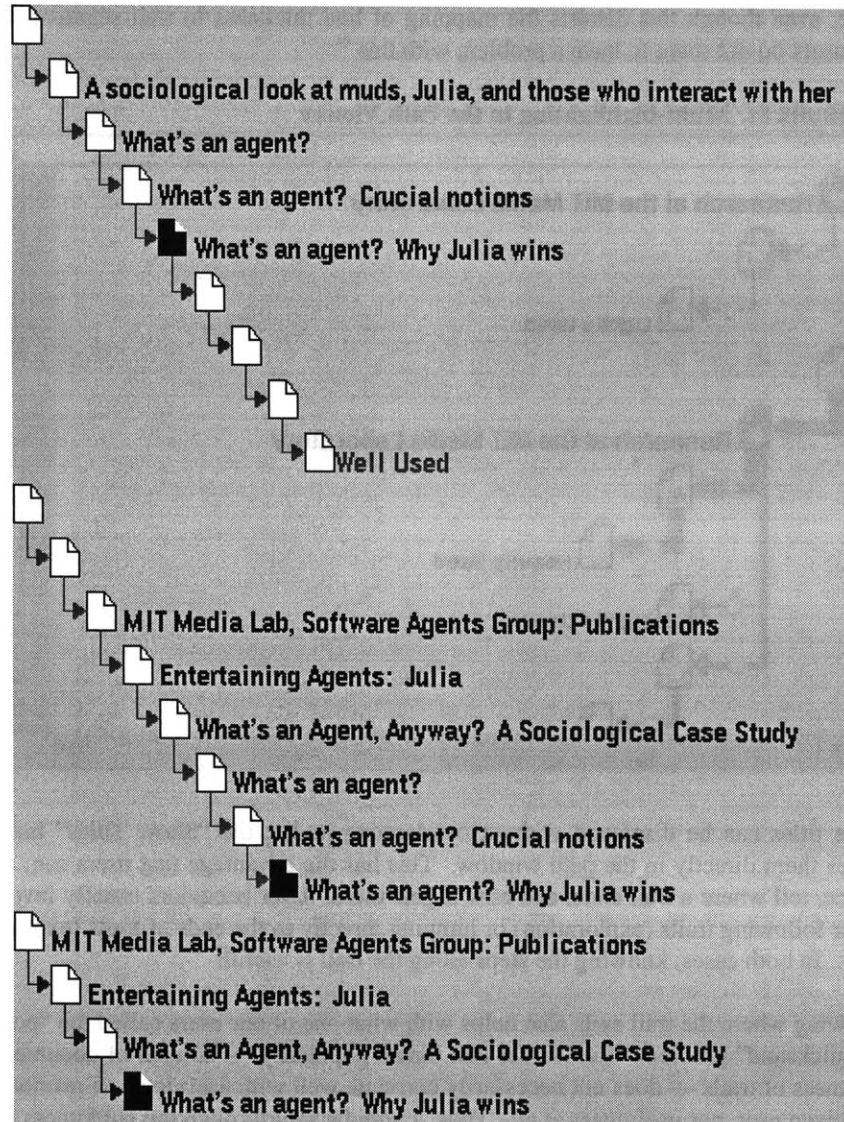
Since titles can be displayed without overlapping nodes, the “Show Titles” button places them directly in the path window. This has the advantage that users can, at a glance, tell where a trail ends, and how it gets there. User behaviors usually involve either following trails (exploration) or jumping directly to the ends of trails (exploitation). In both cases, knowing the steps along the trail is useful.

Knowing where the trail ends also helps with what one of our users called the “popular quicksand” problem. Remember that innate popularity — hotness of documents, thickness of trails — does not necessarily correlate well with usefulness of results for any given user, nor usefulness at all. Thus, a popular trail through the wilderness that led to a pit of quicksand would continue to attract people in a positive feedback loop, though obviously the end result is useless.

6. Of course, it is possible they do have a problem and simply are not telling us; but our experience has been that our users are not shy about indicating when they have a problem.

Trails also show different models of how to navigate information. In particular, they are quite useful at helping place information in context, presumably because users come to the data space (Web site) with different needs or levels of interest. Figure 32 below shows three different trails that exist in our database relating to a paper about an agent called Julia. In the Web-based version of the paper, it is broken up into different segments that can be navigated to in different ways. In this screenshot the user is looking at the “Why Julia wins” section of the paper and we can see that there are three paths that people have used to get to this point.

FIGURE 32. Three Views of a Web-based Publication



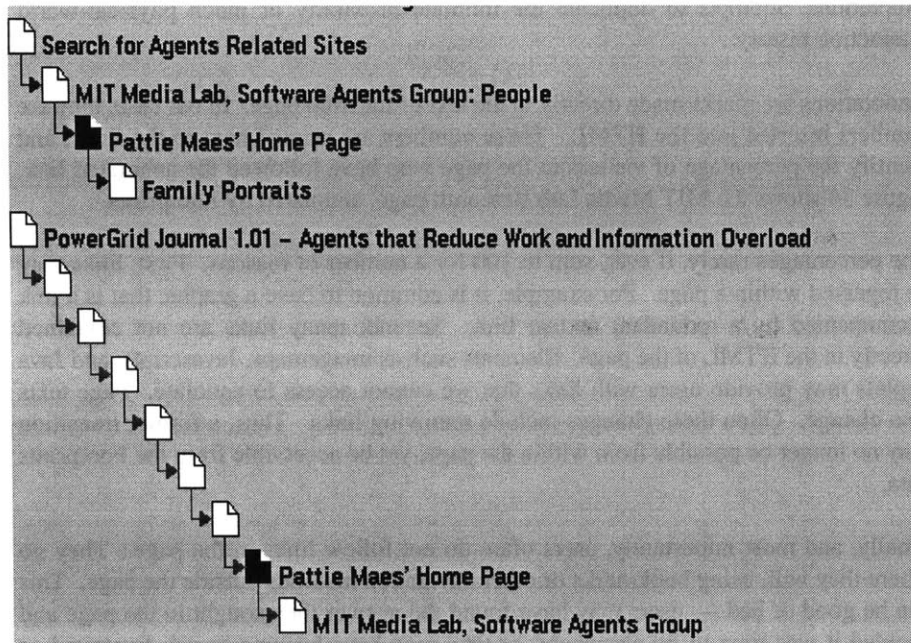
In the least-used trail (bottom of Figure 32) users have started with the Agents group publication page, and skimmed the article, stopping after reading the “Why Julia Wins” page. More users (topmost trail in Figure 32) start elsewhere, presumably are less familiar with the concept of agents, and come to the “Why Julia wins” page only after visiting the page on “Crucial notions.” These users then go on to other pages. Interestingly, the largest group (highest percentage) of users who visit this page come

in from offsite, read several sections of the paper then stop (or go elsewhere). Note that there may be other users who visit other sections of the paper but skip this one. This example is interesting not because it shows anything particularly profound, but because it illustrates how three different models of reading this paper can coexist. We all “know” that writing papers on the Web is not like writing them in text, but as far as I know there has been no attempt before now to capture different ways that Web-based presentations are actually being read. Interaction history gives us a way to see these models and possibly manipulate them.

The next example, Figure 33, shows an interesting dichotomy in how people make and use Web pages. It shows two paths that include the home page of Pattie Maes, who heads the Software Agents group. In the more often used path (the bottom of Figure 33) people start at the “PowerGrid Journal” page. This journal has reprinted an article written by Maes, and in the Web version of the article, each page is given its own Web page. People who make it through the article may be curious, and follow a link from there to Maes’ home page. Since their interest is in software agents, they then go on to read about the Software Agents Group by visiting our group home page.

Reference 47

FIGURE 33. Who is Pattie Maes?



A different group of people start off searching for sites about software agents and for some reason find themselves interested in the people who make such agents. They then follow a link to a page on our Web server that describes the people in our group, including Maes. These people, once they have gotten to her page, continue to be interested in her personally and visit her page of family pictures.

One of the hardest problems in classical information retrieval is the question of deciding what a document is “about.” Various solutions, such as keywords, abstracts, and machine-generated summaries have been tried, each with varying degrees of success. The more homogeneous a document is, the more likely these methods are to succeed. Home pages on the Web are a uniquely bad data set for this kind of technique, because they are so diverse. Even within an individual’s home page there are likely to

Reference 63

be a number of sections about quite different things. Finding a single thing that a home page (or even most Web pages) is “about” is likely to be a recipe for failure.

Interaction history solves this problem by allowing multiple senses of “about” to coexist. In Figure 33, we see that Maes’ home page is in some sense about Maes-the-researcher and, in some sense, about Maes-the-family-person. This multiplicity is natural in our daily lives where we play multiple roles, sometimes simultaneously. Objects may have multiple uses: books can be doorstops or window props or even hammers if necessary.

In addition, these multiple senses can aid our understanding. Earlier, I argued that interaction history was valuable in part because it helped to provide context for information to which users can navigate. Figure 33 provides an example of this: we understand much better who Maes is if we can see her not only as the leader of the Software Agents Group, but also as a person.

3.11 Annotations

Each of the first two tools, maps and paths, has brought the history information closer to the data — made it more proxemic. The third passive history tool, which we call annotations, attempts to duplicate the intimate proximity of much physical-world interaction history.

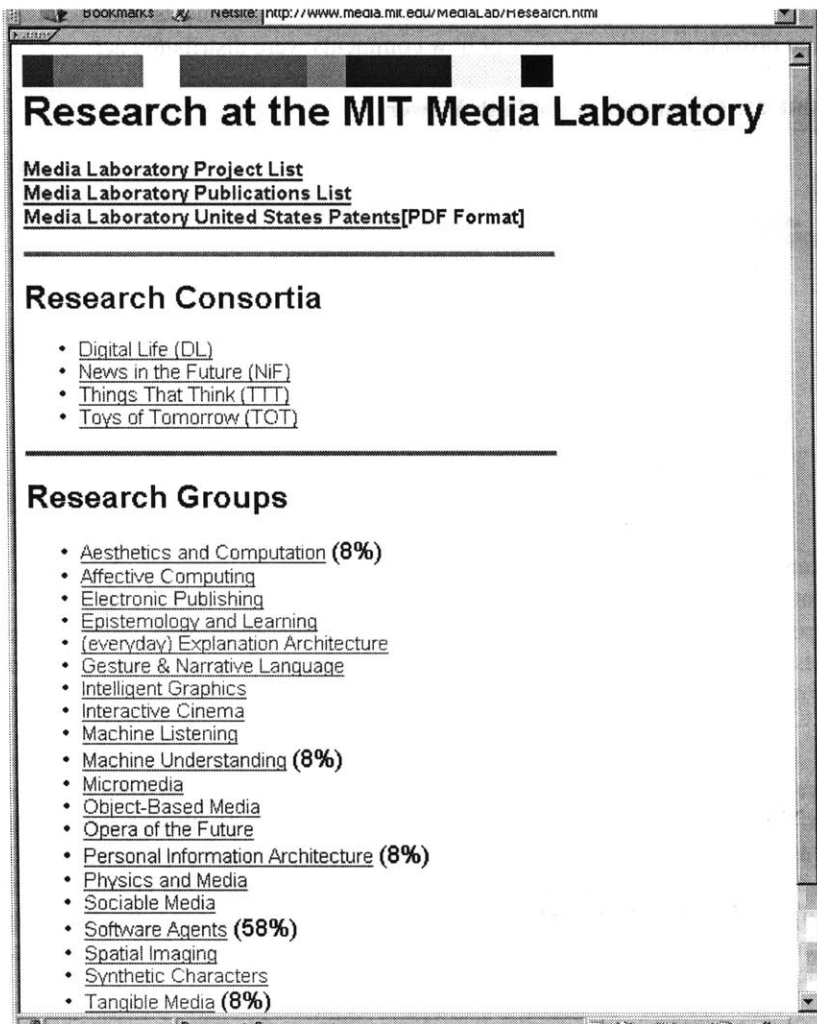
Annotations are marks made directly in the text of the Web page. In our case, they are numbers inserted into the HTML. These numbers are printed next to the links, and identify the percentage of visitors to the page who have followed the annotated link. Figure 34 shows the MIT Media Lab Research page, annotated by Footprints.

The percentages rarely, if ever, sum to 100 for a number of reasons. First, links may be repeated within a page. For example, it is common to have a graphic that is a link accompanied by a redundant textual link. Second, many links are not contained directly in the HTML of the page. Elements such as imagemaps, Javascripts, and Java applets may provide users with links that we cannot access to annotate. Page texts also change. Often these changes include removing links. Thus, a former transition may no longer be possible from within the page, yet be accessible from the Footprints data.

Finally, and most importantly, users often do not follow links in the page. They go where they will, using bookmarks or other navigation methods outside the page. This can be good or bad — users may have found the answer they sought in the page and decided it was time to go elsewhere, or they may have become bored, frustrated or simply decided it was time to look elsewhere.

Because of these limits, annotations also continue the trend from the first two tools of showing less of the total available history information. Trails will likely exist that contain the current document but are not seen in the annotations.

FIGURE 34. Footprints Annotated Web Page



So far all the tools have focussed on the passive side of the spectrum. However, Footprints also includes two active tools: comments and purposes. These tools allow members of the user community to communicate indirectly with each other. Passive history is, of course, a form of communication but it is not adequate for all purposes. As noted in Chapter 2, the kind of history we want to capture depends on the purpose it should serve.

Active History

See "Property 6" on page 27.

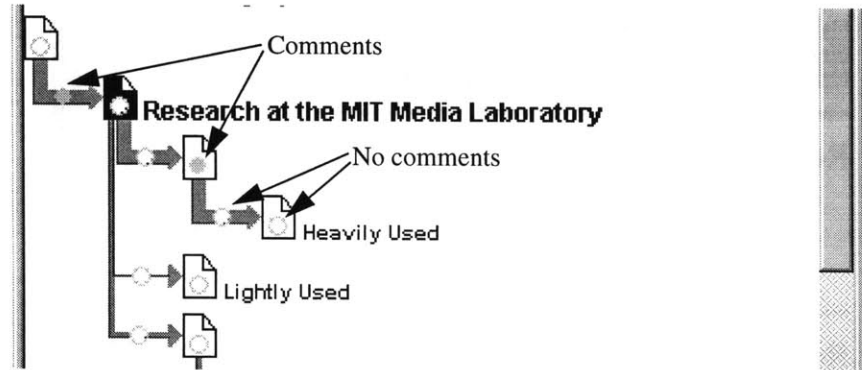
For our active tools we chose to try to provide means to answer the question posed earlier: Is this material any good? There are two ways such a question could be answered. The first would be to attempt some kind of analysis of the content of the pages. However, this is a difficult problem in most cases; furthermore, even a successful analysis would not necessarily tell us anything about the utility of the page. A simpler answer is to allow users to express directly their pleasure or displeasure with what they see.

Footprints provides users with the ability to enter text comments, which are represented by circles on the trails and documents of the path viewer. Circles are empty when no comments have been made, and grayed in when a user enters a comment.

3.12 Comments

We chose to represent commentless pages and trail segments explicitly because we wanted users to have a target on which to click for adding comments and because we wanted to provide a quick overview of how comments were distributed across a view.

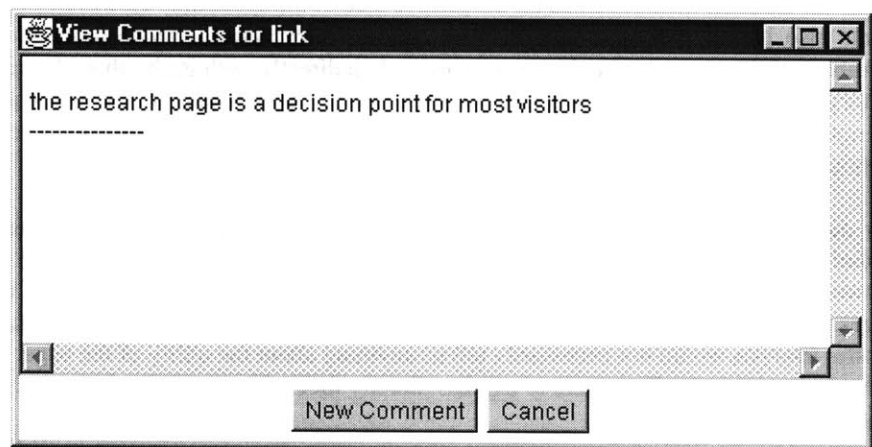
FIGURE 35. Comments in the Path Viewer



Users can comment on documents, of course, but more interestingly they can also comment on the trails themselves. The ability to comment on trails is important for two reasons. First, it helps with our navigations metaphor. In essence, users can leave signposts on the trails; for example, *Go this way to learn about software agents, that way for artificial life.*

Second, it helps users think at a level above that of the individual document. One of our users commented that after using Footprints he realized that the Web on its own was like "looking at the world through a cut out cereal box." By this he meant that in a conventional Web browser one looks at a page, then at another page, and so on; but there is no sense of connection between the pages. In the hypertext community, this has been called a "metadocument" — that is, a "document" that conceptually or organizationally spans boundaries imposed by the information design or by the limits of the software.

FIGURE 36. Comment Window

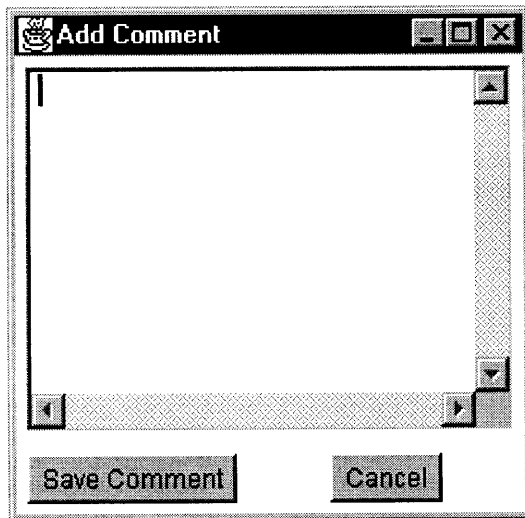


The comment window (as shown in Figure 36) gives a view of the existing comment text, and provides a "New Comment" button for users to add their own thoughts. This window is brought up by users right-clicking on a circle. If the circle is empty, a

blank comment window is shown; if the circle is filled, the window contains all the comments entered for that document or trail segment.

The add comment window (shown in Figure 37) allows users to type their comment. Saving it causes the comment to be added to the database; if this is the first comment added for a document or trail segment, the appropriate circle on the path view changes color, .

FIGURE 37. Comment-entry Window



In our present implementation, we restrict comments to simple text strings. Comments are all stored and all publicly accessible. No comment is ever deleted — new comments are simply stacked on top of existing comments. In a situation where comments were being entered on a regular or large-scale basis, these simplistic techniques would likely be quickly overwhelmed. However, this is not the model our users follow. Comments in Footprints are not like chat; they are an exceptional event. Footprints' users follow the long-established behavior patterns of systems in which users are asked for ratings — they don't give them.

In particular, experience with early versions of systems such as GroupLens — which asked users to rate Usenet news articles — showed that fewer than 5% of users will enter ratings or comments. Those users who do enter information are of two kinds: the early adopters and the passionate. The former will use technology for the sake of the fun inherent in new gadgetry. These people also tend to take the technology seriously, and their comments tend to be the most thoughtful.

Reference 60

The second kind of person who makes comments is one who has a strong opinion. These are often people who have just had a particularly good or particularly bad experience. They want someone to know about it and in some cases they will take the time to record that opinion in a comment.

These two kinds of people are precisely the kinds we would like to have making comments. The general "it was OK" kind of feedback is not useful here. Instead, we want people with care or people with passion to be able to communicate the value of their experiences. Having only a few comments is not, for Footprints, a bug — it is a feature. A large number of comments would only make it harder to find the valuable remarks.

3.13 Purposes

The final Footprints tool attempts to give users the ability to answer the question of “why?” Specifically, it can be useful to know why someone was browsing. In our system, we consider paths to be the fundamental unit of connected foraging, so we augment the page viewer again.

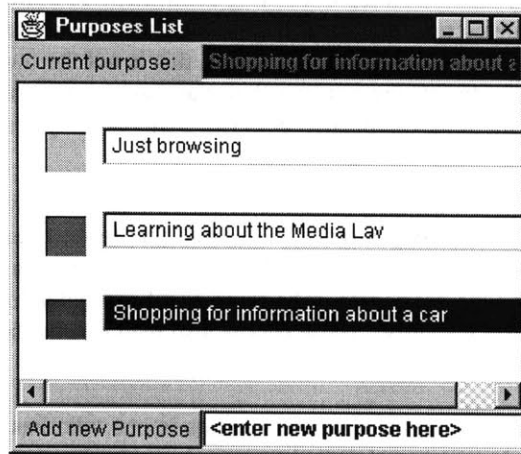
This time we added the notion of “purpose” to the trails, the idea being that the purpose will capture the user’s reason for doing something (if there is a specific purpose). The default “just browsing” is assumed; users do not have to specify this purpose.

Purposes are represented in the path viewer by colors added to the trails. The creation of purposes is at user’s discretion, so colors are assigned randomly at the time a new purpose is created. New purposes and colors for them are communicated to the back end and redistributed to other running front ends.

Purposes are recorded as the user browses in the log files produced by the proxy server. When these log files are processed by the back end, the purpose information is retrieved and applied to all relevant trails. Trails may have multiple purposes; as different users apply purposes, their inputs become votes. The trail is colored on the display according to the most heavily weighted (most popular) purpose currently known for it. In case of a tie, the most recently voted purpose is shown. This is based on the theory that documents on the Web change, and the most recently declared purpose is most likely to reflect the current contents of the documents.

Purposes are presented to the user in a selectable list, shown in Figure 38. Users who find a purpose sufficiently similar to their current intention can click on the purpose in the list. Alternatively, a text entry field is provided at the bottom of the window for users to enter new purposes.

FIGURE 38. Purposes Tool



As with comments, newly created purposes are entered directly into the database. Users who apply a purpose have the effect of assigning that purpose to any trails they follow from that point on. The assigned purpose stays in effect until either the user explicitly changes it or — more often — until the system estimates that the user has changed tasks. This detection is performed in a manner similar to the way trail segmentation is done. Specifically, when the system detects that the user’s dwell time is too long, then the purpose is turned off — returning to the default “just browsing” purpose.



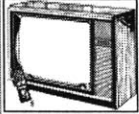

One difference, in this case, is that we are not measuring just the time between page loads. We track the time between user actions on any of the Footprints tools. If the user is not taking any action and is not looking at any new pages, the system assumes the purpose has changed. When the system reasserts the default purpose, the user is notified via a pop-up window and has the opportunity to tell the system that the former purpose is indeed still valid.

As with comments, few of our users take the time to assign purposes, despite it being the single most-requested feature in our email feedback. Ideally we would have liked to have a method for assigning purposes automatically. However, this is almost impossible, given the constraints of page variability and the limits of information retrieval techniques.

We attempted a simpler solution on the Footprints main Web site, as illustrated in Figure 39. The home page contains a “Paths” button that takes you to a different page with a table of all the paths in the Footprints database. These paths are organized into loose groupings of what they are “about,” as determined by the keyword computation process described below.

Footprints Paths page

FIGURE 39. Paths Page

	computer disk problem	3	Lightly Used
	tank army war	4	Lightly Used
	television media broadcast	5	Lightly Used
		3	Lightly Used
		3	Lightly Used
		3	Lightly Used
		3	Lightly Used
		3	Lightly Used
		11	Lightly Used
		3	Lightly Used
		4	Lightly Used
	secret agent assistant	4	Lightly Used
		11	Lightly Used
		6	Average
		4	Average
		9	Heavily Used
		3	Lightly Used
		4	Lightly Used
		3	Lightly Used

The ‘aboutness’ of paths is shown in two ways. One (shown on the left of Figure 39) is an icon intended to be evocative of the path. The other (in the center of Figure 39) is three keywords that relate to the icon and help users understand its meaning; for example, the picture of the man wearing a hat is identified as “secret agent assistant,” indicating that the trails associated with that icon have something to do with agents. Keywords are created by downloading the text of the documents associated with the trail and analyzing them, as explained below.

The next field in the table gives the length of the trail, so users know approximately how much data they may be looking at. Finally, the last field serves two purposes. It tells how used the trail is (in the same four general categories as in the path viewer — light, average, heavy and very heavy), and also serves as a link to the first node in the trail, so users can click once to go directly to the start of the path.

Reference 63

Keywords for paths are generated by a simple algorithm that is a variant of the familiar TFIDF (Term Frequency Inverse Document Frequency). That formula is useful for determining keywords that best distinguish a document from a collection of similar documents. In our case, though, TFIDF does not work because we want the words that are typical of the whole collection. TFIDF is computed by the formula:

$$\text{TFIDF}(w, d) = \text{TF}(w, d) * \log(\text{numDocs}/\text{DF}(w)) \quad \text{(EQ 2)}$$

That is: the *tfidf* value for word *w* in document *d* is the term frequency *TF* (the number of times ‘w’ occurs in document ‘d’) times the log of the number of documents divided by the document frequency (the number of documents that contain *w*).

However, the *log()* term means that as the word *w* is found in more and more of the documents, the term approaches *log(1)*, which is 0. This is precisely the inverse of what we want, since words representative of the trail as a whole (rather than of any individual document) should appear in most of the documents in the trail.

Instead we use a simple formula that I call my “cluster characterizer” (*cc*):

$$\text{cc}(w, C) = \sum_1^{\text{nD}} \text{TF}(w, d) * \text{DF}(w)/\text{numDocs} \quad \text{(EQ 3)}$$

Thus the *cc* for word ‘*w*’ in document cluster ‘*C*’ is the sum of the term frequencies for ‘*w*’ in all documents in the cluster times the percentage of documents in which this term appears. This equation has the behavior we desire, as terms that appear more often in documents are weighted more heavily. We remove common words (and, or, the, etc.), stem the words (removing endings such as ‘ing’ and ‘ly’), and take the ten keywords with the highest *cc* scores for each trail. These keywords are then matched against the hand-coded keywords for each icon to determine the appropriate icon, and the top three keywords are shown to users.

Footprints tools in relation to theory

As noted at the beginning of the chapter, the set of tools built for this project is not intended to be the ultimate or even the sole set of possible tools for aiding Web navigation. Instead, the set of tools built for Footprints serves to illustrate different points in the design space for history-rich systems, allowing us to explore possibilities. The following diagram illustrates points in the space that have been explored.

Figure 40 summarizes where the five Footprints tools fall within the interaction history framework. Each property of the framework is drawn as a line, labeled with the elements of that property. So, for example, proxemic and distemic are at opposite

3.14 Summary

The maps, paths, annotations, comments, and purposes give some coverage to the space. Obvious gaps exist in areas such as personal history tools. This chapter has attempted to give a structure for talking about navigation in terms of information foraging, including elements of both exploration and exploitation. These activities take place in an environment of user's models of how information should be organized. The basic elements of such organization are similar to Lynch's basic elements. We add to that the notions of sequentiality and centrality, to account for the way that Web information is experienced with current browser software.

This chapter also described the data structures and internal architecture of the Footprints system. Using a proxy server and standard Web logs, two basic data types — the extended transition and the path — are constructed. These are built into trails and clusters, which are used to display the interaction history information on, and next to, the Web page being viewed by the user.

All paths lead nowhere, so it is important to choose a path that has heart.
Carlos Castaneda

Reference 10

Having described our particular theory of navigation as an information foraging activity, and then implementation of a particular set of tools responsive to our needs and to the desires of our users, this chapter discusses related implementation work — tools, prototypes, and implemented systems that relate to Footprints in interesting ways. In particular, we will cover work that touches on issues raised by Footprints or that presents alternative ways to interpret the theoretical approaches we used. We will deal first with work related to interaction history, then sample work in social navigation systems. Beyond that, we will touch on guided navigation systems and Web trail systems.

Implementations covered in this section are generally targeted at problems different from those tackled by Footprints, but are related in that they have a common theoretical basis. In particular, all of the first set of these systems descend from a common ancestral idea.

This idea is captured in Hill and Hollan’s initial work on history-rich digital objects for software project development. Their tools, primarily Edit Wear and Read Wear, were oriented toward showing the history of use of source code files in a software development project. The notion of “wear” attempted to literally invoke the wear and tear that affects physical objects as they are used.

The methods used in their tools for showing “wear” involved augmenting user interface widgets. For example, the “thumb” in the scrollbar of their tools was thickened to show the amount of time spent by users looking at a given region in the file — a concept similar to the notion of dwell time. In the Edit Wear tool, scrollbars also contained a series of markings that encoded different useful historical properties such as where edits were performed, how long ago, and who made the edit. Other tools allowed these marks to be connected so that, for instance, a section of code copied from elsewhere in the source tree could be traced back to its original site. This made software tasks, such as requirements tracking and debugging, much easier. The source code itself was also enhanced to show the accumulated changes so that change histories could be seen in summary. This allowed easy identification of “hot” areas in

4.1 Related interaction history work

The Wear tools

Reference 34, 35

See “Dwell Time” on page 65.

the code suite. These hot areas tracked where development was ongoing and what parts of the code were the most heavily referenced, a sign of importance and sometimes of difficulty in understanding. Email was also enhanced to show operations performed on it such as last time read, whether or not it had been replied to, and so on.

The Wear tools were all social — that is, used by a project group — and one of the problems discussed by the authors is what effects might come from taking normally private actions such as reading an email message, even one sent to a group, and making them public. The Wear tools all attempted to make the information as proxemic to the users of the original data as possible, but it was still necessary to learn a number of rules for interpreting the new scrollbars and other widgets. Hill and Hollan emphasized the attempt to mimic the way interaction history works in the physical world. This did not lead them to attempt to permeate the history information into the data, though. In fact, their “Vita Service” tool implemented an abstract layer that captured history information as first-class elements for inclusion in other tools, thus maintaining a separation between the data and its history. In the Wear tools, history changed at the rate people changed the underlying data. No information fading seems to have been included.

Pad++

Reference 4

Hollan, in collaboration with Ben Bederson and others, has continued to pursue ideas related to interaction history. In particular, the Pad++ system was constructed to form a testbed for what the system’s authors call “alternate interface physics.” The notion of such a “physics” would be to attempt to mimic the forms and processes that lead to wear on physical objects, but in a digital context. That is, by mimicking physical processes in a digital environment, interaction history traces can be made to appear on or with Pad++ objects.

Pad++ is a system on which a large number of applications can be built. These applications can do things like drawing, presentation, computation, and so on. The underlying Pad++ layer provides persistence for objects and attempts to smooth user access to stored information. The Pad++ underlying metaphor is that of an infinitely large, infinitely stretchable sheet. Objects in a Pad-based application all have an absolute location, computed based on a system notion of the origin. The relative positions of any number of objects can be computed easily and, because the underlying “sheet” stretches infinitely, it is always possible to place objects at any relative location, such as between two formerly adjacent objects.

Pad++ supports navigation by panning around the displayed surface. Users can also zoom into or out of any place on the display. Zooming can either or both enlarge objects so they can be examined in more detail or reduce detail so that users can quickly navigate to distant locations. More interestingly, though, Pad++ offers an operation that is called “semantic zooming” — scaling data in nongeometric ways. A semantic zoom does not per se change the size or apparent size of the viewed objects; instead, it causes more (in) or less (out) information to be displayed. For example, a clock display might start off showing the approximate time and when zoomed in semantically, might show first the precise minutes, then seconds, and so on until the maximum resolution was reached. Semantic zooming can be used as a form of information retrieval. For example, zooming into a headline can bring up more detailed information, such as a news story and a full-sized image associated with the headline. Zooming can initiate computation; in this sense, semantic zooming is something of an umbrella metaphor for actions that are common in standard user interfaces but are currently scattered around a number of different command interfaces and widgets.

Several different HTML viewers have been implemented in Pad++. The underlying capabilities make it easy to design different layouts for retrieved Web pages, such as tree structures showing which pages can be reached from which other pages. In addition, semantic zooming capabilities make it possible to represent relationships among pages. For example, zooming into a page could reveal a layout of all the pages that point to this one, something that is easy to compute but that is rarely shown in Web browsers.

Reference 5

Bederson argues that we need different styles of interface, citing simulation-building environments such as Smith's Artificial Reality Kit as examples of ways that physical models can be digitally simulated. Although simulations provide natural ways to integrate history into digital information, they do not address the general problem of associating history with digital data nor give guidance on what history can be used for.

Reference 74

Reference 73

Since Pad++ is instantiated in so many different applications, it is hard to know how to apply our framework to analyze it. Seen from the point of view of the authors' goals and the system's potential, Pad++ could actually be the most proxemic history system possible, if interface physics are designed that actually mimic real-world physics. It remains to be seen whether this would be a good idea, however. Pad++ itself is not a social system, but it allows users to easily incorporate information from multiple people and multiple sources. An application could be designed that gave people access to a common "sheet" of information. Depending on how a semantic zooming operation was designed (or seen), it would be possible to make history information fully permeate, or reside separately from Pad++ objects.

Pad++ could also include passive history capture mechanisms, in much the way it now passively handles data storage. Similarly, a facility could easily be designed that would allow applications or users to control the rate of change; currently, objects change either in response to user actions or to computations spawned by user actions.

Other work has also drawn on the original Edit Wear and Read Wear ideas, and implemented explicit interaction-history mechanisms. For example, Dieberger has created a history-rich MOO called Juggler. A MOO is a form of multi-user virtual environment, based in text. Locations, objects, and people in a MOO exist as code in a server and are expressed descriptively with lines or paragraphs of prose. For example: *you see a book. It is old, the cover is tattered, and the binding is starting to fray.* Users can then take actions on these objects, such as *pick up book*. MOOs have a basic concept of place called a room, and these rooms are linked together by passageways or other means of access.

Juggler

Reference 15

Users gather in rooms for particular activities, such as private chat, playing games, or listening to public presentations. MOOs are fundamentally social in nature because they serve as meeting places. They can also allow their users to customize or program the system. New objects can be created, and new rooms be assembled and linked to the existing network of rooms.

Normally, traversal between rooms is by these connections — the underlying MOO software annotates room descriptions. So, for example, a user might be told while in the *dining* room that the *kitchen* is to the *north*. Then typing the command *north* would move the user to the *kitchen* room. Juggler enhanced this navigation paradigm by keeping track of the use of the transitions between rooms. Exits from each room were labeled with the relative degree of use they had received. Users could find out

which exits had been most heavily used but, as in Footprints, were not obligated to follow the most-used paths.

Objects within the Juggler MOO also were history enriched. In particular, communications objects such as bulletin boards were enhanced with read wear. Newly added messages were shown as “new” with both a text annotation and a red color. As messages aged, their color faded toward gray. Whenever the message was read, the text tag also changed to say “recent.” This use faded over time to “normal” and eventually to “old” for messages that had not been read in some time.

Interestingly, Dieberger also discovered the problem of monotonically accumulating history information. His imposed fade function simply halved the use count of each room exit every two weeks, which seemed to work well.

In the terms of our framework, Juggler is proxemic, since the history mechanisms use the same techniques as the underlying software. It is also highly penetrated — there is no separation between the history and the underlying data. MOOs are social environments, and Juggler’s history is also social. Finally, it is mostly passive: changes happen without users doing anything different from their normal activities; however, object designers in the MOO (who are usually the same as the users) must actively code interaction history into their objects. Objects may also inherit history behaviors from their ancestor objects.

Patina

Reference 67

Within the Media Lab, Arjan Schütte’s Master’s Thesis explored the visual display of wear. This system, called Patina, was inspired by the appearance of patina — or visible color change — on physical objects such as brass doorknobs. Schütte’s concern was not with the theoretical bases or problem-solving capabilities for interaction history, but rather with the integration of interaction-history effects into a coherent narrative interface presentation.

The Patina map, as his tool is called, involved using a programmatic wrapper around a graphical image in the form of a Web page imagemap. An imagemap is a picture presented in a Web browser that is sensitive to where a user clicks. When a mouse click is detected, the X,Y coordinates of that click are sent to the Web server, which maps the location to an area and may invoke a program in response. In the case of Schütte’s thesis application, the image was a map of Thailand, part of a Web-based presentation of his work there. The map organized the text and images of the presentation into a spatial framework. Users could navigate from page to page by means of links in the text, but could also navigate to new clusters of pages by clicking on the imagemap.

The Patina map recorded where users clicked and showed the results visually on the map itself. The effect was a form of wear similar to what might have happened if the map had been printed on conventional paper and subjected to use by human hands. The image was both degraded and enhanced. It was degraded in the sense that bits of the original were obscured by the Patina software, which had the effect of reducing the legibility of the image. At the same time, however, it became easy to see at a glance where the important points, items of interest, and major story elements were located on the map.

Patina used color as another way of conveying history. Newer text was presented in red and slowly changed hue towards blue as it aged. Darker, more saturate colors were used to signify greater data density — more heavily traveled paths, and more visits to sites.

Schütte differentiates three properties of interaction history traces: the medium of its existence (physical versus digital), the method of its delivery (local versus remote, asynchronous versus synchronous, foreground versus background), and its granularity (deep versus shallow, intra-session versus inter-session).

See "The trace" on page 25.

Patina's idea was that history information could be gathered into and expressed as a layer on top of the conventional layers of digital information. Schütte also applied the ideas of Patina to other storytelling situations. In these situations, the layer supplied by Patina did not change the underlying image. Instead it provided a layer of use-based information that helped guide people along different possible story paths. By flashing an icon on all Patina maps whenever any user clicked on any map, Patina allowed users to be aware of others' presence in real time; however, there was no way of knowing who the other users were or communicating with them.

Patina shared our concern with users' experience of the information. It attempted to track these experiences by changing the effect of clicking on a link contained in a Patina map. These links brought up the target document in a new browser window, which also contained an applet that monitored users' further progress.¹

In the terms of our framework, Patina attempts to be proxemic by mimicking the notion of a patina on physical objects. However, the rules by which the Patina information layer is applied are still arbitrary and must still be learned. The history does permeate the Patina map, but not the information that is linked from the map. Unfortunately, the basic notion of history as a separate layer seems contrary to the possibility of permeation, though the actual implementation of the system makes the history seem to permeate the display. Patina is social, in that others' information is visible as part of the map; seeing where others click is not per se a social interaction history technique. It is also passive, since all history is created as a by-product of user actions. Finally, the rate of change of Patina history is not specified; it appears to simply accumulate.

An approach that seems superficially similar to Patina is the Live Web Stationery of Seligman and Bugaj. This system most closely resembles the initial implementation of Patina, but with a more abstract artistic bent.

Live Web Stationery

Reference 68

Where Patina had used a simple bit-wise scheme for altering images, Live Web Stationery uses a complex set of image manipulations to convey interaction history. These changes to the images give the effect of treating a Web page as though it were a piece of paper. A page in the Live Web Stationery system show smudges, rips, stains, and fading. Live Web Stationery applies these modifications to the image of the page, based both on human activity and on a model of how paper reacts to use.

Live Web Stationery works by giving the user a modified browser software. This software does not display HTML directly, but rather synthesizes an image. This set of pixels is composed from the pixels of the original Web page and various pixels created by the Live Web Stationery system based on history of use. Thus, a Web page may appear to have "creases," as if it were a piece of physical paper that had been folded and unfolded. The modified browser performs the same functions as a stan-

1. We had considered a similar strategy for Footprints but had rejected it in part because of reliability problems with applets and in part because such a strategy is impractical when a system must deal with a wide variety of different possible Web pages instead of a constrained set used with Patina.

ard Web browser, such as fetching Web pages. Based on the published description, it is not clear how the system operates technically.

Unfortunately, Live Web Stationery suffers from what, in our framework, would be called extreme distemia. The correlations between the image manipulations and the history they represent are obscure at best. A coffee stain may — as implemented in their system — represent the fact that the page has been read many times; but unless someone tells you this rule, it is hard to know whether anyone could intuit the connection between use and the trace shown by the system.

Live Web Stationery also has not been deployed beyond an experimental proof of concept. The Metaphorium project — of which Live Web Stationery is a part — has a number of possibilities for embedding history into different systems, but seems to lack a rigorous basis for helping users understand how the history is being used.

Alexa/What's Related

A similar problem affects Alexa, a browsing recommendation system. This commercial product runs as a tool in the background on a user's PC. Whenever a page is loaded into a Web browser, Alexa attempts to provide a list of pages that are related to the current page. In part, the selection of pages provided is based on what links users followed from the current page. Alexa's tools track what pages are accessed by users as they browse and, in some way, communicate that to a central server for inclusion in the history database.

However, Alexa's recommendations are not based on history information alone. As a piece of software for which users have paid money, it is not going to seem useful if it cannot provide recommendations when needed. Given the size and changeability of the World Wide Web, and the slow rate at which history accumulates, Alexa cannot rely solely on history and still be certain of being able to provide recommendations at all times. Although the company has not published its techniques, it appears from inspection — that is, by using their software — that they make use of at least two other methods for making recommendations.

The first is keyword matching. Keywords in the current page may be matched with keywords in documents already known to Alexa, and pages with sufficient numbers or strengths of match may be provided as recommendations. The second possible technique is site similarity; that is, Alexa appears to assume that if two Web documents come from the same net domain, they are similar. Thus, when I look at my home page with Alexa software, it suggest other pages at mit.edu as related, although inspection of their contents shows no similarity between the suggested pages and mine.

Recently (around the summer of 1998) Alexa licensed its technology to Netscape, which incorporated it into their version 4.05 browser software as the "What's Related" feature. This feature appears as an additional button on the browser control panel that, when pressed, pops up a list of recommended pages, related to the page currently being viewed. The browser software also sends back the pages being viewed to its database, so that history will continue to accumulate. Unfortunately, many people using the feature were unaware that their traversal data were being collected simply because they turned on this feature. Although the user interface does not present any connection between the links suggested and any individual who might have led to their being shown, a natural worry remains about what is being done with users' traversal data. The amount of personal information collected and associated with the history information in the browser software is unknown. Finally, the blending of history and other techniques may lead to more frequent or better recommenda-

tions, but it makes the effect of history much harder to detect. The result is that people feel their information is taken without any perceptible payback.

In the terms of our framework, Alexa/What's Related are highly distemic in their construction but more proxemic in their use, since interaction with the recommendations follows the same paradigm as interaction with URLs in a Web page. The Recommendations have a social aspect but the community is hidden from the end users, even though they participate in it. If there is any personalization of the history being done, it is out of sight. In my informal experimentation with Alexa I was unable to detect any change in the items being recommended. What's Related, on the other hand, seems to change on an almost daily basis. My guess is that this is due to the large amount of input being sent to the database by all the Netscape browser users. Unfortunately, this rapid change makes it hard to rely on these recommendations. The history, insofar as it can be seen, is completely separate from the data.

Matthew Chalmers and his colleagues have been working on a series of systems, of which Recer is the latest, that use history in a much more direct fashion to provide recommendations. Their systems operate by monitoring changes to a Web browser software's history file. Recer performs a path-based recommendation analysis by looking at changes in this file.

*Recer: Activity-Centered
Information Access*

Reference 11

The basic idea is very similar to that of Footprints, in that if the system detects that the user has started down a path, it suggests that continuing to follow the path might be fruitful. Users can consult a window adjacent to their browser to see URLs that the system wishes to suggest. Recommendations are made based on two techniques: path segment overlap, and URL co-occurrence. The first technique is similar to Footprints' method for deciding if two paths are part of a larger path. In essence, if a user's actions overlap with part of an established path, then the rest of the path is assumed to be valid and may be presented as a recommendation. The second technique is order independent, and simply compares groups of URLs as if they were data sets. If the user has already visited a sufficient number of the URLs in the history set, then others in the set may be provided as recommendations.

Recer has apparently not been tested by users other than the system authors, and many of the techniques and interfaces used by the system are still under evaluation and change. For example, recent work has involved implementing an activity-watcher and recommender for the *xemacs* text editor (Chalmers, personal communication).

In terms of our framework, Recer is distemic in much the way that Footprints tools are distemic. The recommendation information can be extracted from paths gathered from the data of a number of users, so it is potentially a highly social system. The rate of change is the rate at which users browse the Web, and it is a passive history system. Users cannot directly contribute to the history information, except by changing their Web-browsing patterns.

Social navigation systems are tools that help people deal with information overload problems by providing guidance and/or recommendations. They often do so in a content-independent way. Rather than examining the data of an item (such as text keywords) and determining similarities of data contents, social navigation uses the activities and opinions — often in the form of ratings — as guidelines. Using ratings usually requires embedding into the system relationships that mimic general social conventions. For example, GroupLens assumes that if people agreed in their past rat-

4.2 Social Navigation Systems

References 16, 61, 70

ings then they are likely to agree in their present opinions; HOMR assumed that if users had similar likes in musical groups then they would also have similar dislikes. Even though these assumptions may be false for any two individuals picked at random, they are likely to be true for individuals selected from a large pool of participants. This is similar to Footprints' assumption that while the individual history of any one person may be only marginally useful for any other person, the collected history is much more likely to contain useful information.

See "Alternative Model of Navigation" on page 46.

Social navigation systems are related to the Footprints tools in that they often take a similar approach to helping people find useful information in a complex information space. Footprints takes social navigation to be a paradigm useful for solving the kinds of information foraging problems in which we are interested. However, we can also look at the core datum of social filtering systems — the rating — as a form of interaction history. Generally speaking, this is a highly active history form; users read articles, listen to music, etc., and give their opinions. Those ratings are usually numbers on some scale and are combined in some way to come up with a ranking for the item.

Most of these systems also share with Footprints what we in the Software Agents Group have been calling the "cast of thousands" problem. That is, in order to have enough data — history, ratings, or whatever — to be useful, the system must be collecting input from a large number of users, potentially thousands of people. Failures of such systems may be due not to weaknesses in the theories or problems with the tools themselves, but simply to an insufficient number of data-providing users. There is no a priori solution to this problem; in fact, many systems make sense only when deployed over a large population. It is something we must keep in mind when evaluating the performance of these kinds of systems.

Tapestry

Reference 28

Probably the first social filtering system was the Tapestry project, developed by Goldberg and Nicholas at Xerox PARC. Tapestry began as a text-filtering system used to deliver data streams such as email or news articles to users' computer desktops based on interest profiles. As part of the profiling system, users could enter judgements on the usefulness of items recommended by Tapestry. These judgements were simple binary like/dislike annotations added voluntarily by users.

Tapestry selected items for users to read based on a two-phase scoring algorithm. For example, the presence or absence of certain keywords could affect the score, as could things such as a messages' author, place of origin, and so on. Once these general rules had been applied by the server, the result was a decision on which data items should be sent to which users. At the user end, the Tapestry client received the stream of items and applied a second, more sophisticated set of rules. These rules produced a scoring, or weight, associated with each item. The weights were then used to sort or rank the items into a presentation order. The goal was first to eliminate items that were unlikely to be of interest at all and second to present the most interesting items most prominently.

Given this architecture, it was a relatively simple step to make users' feedback responses into another rule within the system. User A could set up a rule increasing the importance of any message that had received a "like it" response from user B, or any of a group of users.

Unfortunately, this created a sort of chicken-and-egg problem. Someone had to be the first to read and rate a message. This rating potentially benefited many other people, particularly when one person's judgment was highly valued. But the person who read the message first received little or not benefit; indeed, if there were a large number of

uninteresting messages, being the first to read them would be a negative since one would have to wade through all the data whereas others who waited got a more useful filtered stream. In fact, if the first reader did a good job, the problem would likely become worse because more people would wait to read the messages until feedback was provided on what was worthwhile. In a system that is widely deployed, this problem is less acute, since it is likely that many people will not want to wait and not many will be waiting on data about the same message. In the kind of tightly knit group that used Tapestry it became a serious issue and probably discouraged many people from using — or at least from entering data into — the system.

In our framework's terms, Tapestry history was proxemic in that it integrated directly with the rules, but distemic in that the effect of history on the eventual data stream was opaque, a problem that plagues many of the systems in this section. Tapestry was social in that rules could use other peoples' ratings but also personal in that users could encode actions based on their own ratings. Tapestry interaction history was separate, and faded extremely quickly, since articles did not last long.

GroupLens takes an approach similar to Tapestry, but on a much larger and more persistent scale. GroupLens' target is the vast mass of groups and articles available on the Usenet news system. The number of newsgroups is well over 10,000 and the number of articles posted on all groups every day is roughly an order of magnitude greater. Articles are posted onto the writers' local computer and propagate to other machines by a copy-and-flood protocol. With several high-speed, high-capacity links connecting the essential parts of the network these days, it is rare for articles to take more than two or three days to reach all machines, worldwide. On the remote end, news server programs provide articles to news readers, which are programs running on users' local machines.

GroupLens

GroupLens runs as a layer above the standard news-article distribution protocols, with a set of customized readers available for use. These readers allow users to enter a rating on a 1-5 scale. These ratings are collected by special servers called "Better Bit Bureaus" (BBB) that keep track of related information, such as who issued the rating and in response to what article, as well as keeping a ratings database.

Reference 60

The BBB acts as a packaging facility, collecting relevant ratings and distributing them along with the articles, as a normal news server does. However, the BBB can also make predictions. Given the right information, the BBB can compute a predicted rating for any article for any user. This is done with a form of collaborative filtering.

Reference 70

Collaborative filtering uses data external to the rated object; in its simplest form, collaborative filtering can be thought of as manipulating a matrix of numbers. Each message or other item known to the system is a row, and each column is a user or other source of ratings known to the system. The numbers in the cells of the matrix are the ratings. These can either be directly entered into the system by users or can be inferred automatically. In effect, the past history (users' ratings) is used to predict the future (what users will think of data items they have not yet seen).

How social/collaborative filtering works

Computing the rating is accomplished by finding users who have two important properties:

- They must match up with the target user in existing ratings. Different algorithms can be used for different results. One algorithm might require that users agree on both positive and negative ratings while another only requires agreement on positive rankings.

- The user who matches must have rated the item for which a rating is to be computed. The past users' ratings serve as input to the computation function, so if the purportedly matching user has not expressed an opinion about the item in question, it is as useless as if he did not match at all.

See "Satisficing and interaction history" on page 34.

In essence, social information filtering assumes that if two people agreed on ratings in the past, they will agree on ratings in the future. Although this is not a certainty, it models our behavior in the real world well enough. In this respect social information filtering, like other forms of interaction history, is a satisficing technology.

The problem with GroupLens, like many such schemes, is that when the space of items for which ratings may be known grows large the matrix becomes sparser and it becomes harder to figure out what the predicted rating should be. Statistical techniques can be used to approximate or interpolate missing data but these techniques also become less useful as the space grows. This is a variant of the "case of thousands" problem, in that one possible approach to solving it is simply to increase the number of users in the hopes that with more users there will be more variety and more likelihood of finding a user or users whose data can be fed into the prediction algorithm. Unfortunately, GroupLens has not been deployed widely enough to give any data on whether this will happen. Experience with HOMR, a musical-artist rating and recommending system indicates that this is possible. HOMR's user base grew over the course of its two-year deployment to many tens of thousands of people, and users had little problem finding matching data, even though the space of possible musical artists is quite large.

Reference 69

It is also possible to track differences between predicted ratings and ratings entered by the user. An intelligent or learning algorithm could notice significant deviations from predictions and adapt its predicting behavior. Although GroupLens did not use this kind of adaptation, the statistical techniques built into systems like GroupLens and HOMR contain some corrective properties. If the matching users are recomputed after each new rating is entered, then as dissimilar ratings are placed into the matrix, the formerly matching users are less and less likely to be selected as closest matches, and others are more likely. The algorithm refines its selections as more data become available, giving the impression of learning.

GroupLens, as the name implies, is socially oriented. Although it is obviously necessary to keep an association between a user's system identity (pseudonym) and his ratings, the authors of GroupLens were aware of the potential privacy implications and took steps to hide the true identity of the authors of ratings. They also took steps to make their ratings information more proxemic than standard ratings systems. For example, one of their client software systems, aimed at use by colleges students, changed the ratings inputs from 1-5 to A through F on the assumption that this would be more familiar to students than a numeric score. Ratings in GroupLens do not permeate the text of the articles, but they do permeate the summary displays afforded by most news readers. Usually such displays have one line per message, showing the article's author and/or subject; GroupLens tools added a field showing the predicted rating for the current user for that article, attempting to tell users whether they will like or dislike each article.

GroupLens history, like most ratings systems, is active. Users must make a conscious decision on how to grade the message. The authors of the tools realized that this could be disruptive, so they created a solution that attempted to move toward passive history. In most news readers, users must press a key to indicated that they are done reading a message and want to see the next one. GroupLens readers were modified so

that selecting any ratings number also had the effect of moving the user to the next message; thus, ratings entry was made more a part of the normal reading process. Finally, GroupLens history has an interesting rate of decay. Normally, Usenet news servers discard messages after just a few days, to make room for the flood of new incoming articles. BBB purge their message data regularly, but retain the history (ratings) information so that users can be better matched. Thus, the history outlives the information.

PHOAKS (People Helping One Another Know Stuff) presents a different approach to providing recommendations. The system uses passive history based on Usenet news articles to recommend Web resources. Specifically, PHOAKS tools analyze messages within a newsgroup, looking for messages that contain URLs. These are then subjected to more detailed analyses to discriminate certain ways in which URLs might be mentioned:

- positive recommendations or announcements;
- negative recommendations;
- Frequently Asked Questions (FAQ) lists;
- homepage URLs, including corporate or sponsoring organizations.

The categorization techniques use two basic kinds of heuristics. One is linguistic, and detects positive versus negative words in sentences containing or around URLs. The second uses message structure, since homepage and corporate URLs are more often found in message headers and footers (called a “.signature” — a short block of text automatically appended to outgoing messages by news posting programs). Once the URLs are categorized, PHOAKS updates its database, noting which URL resource has been mentioned, in what newsgroup, and by whom. Resources mentioned in the database are sorted on a simple popularity basis — a kind of one-person/one-vote system that guards against people spamming the system. The most widely mentioned URLs are selected for each newsgroup and a Web page is created on the PHOAKS Web site. The system also keeps track of (and publishes) the histories of the people making recommendations, giving an aspect of identity to the otherwise-anonymous recommendations. This allows users to go from a recommended URL to the list of people who recommended it and, for each of these, see what pages the person has recommended. This is an excellent balance between personal and social histories. The authors of PHOAKS also continue to be aware of the possible privacy and security implications of their work, although they have not yet implemented any major privacy components in their systems.

The PHOAKS Web site is also history-enriched in other ways. For example, it includes timelines for each recommended URL showing the recency of the recommendations, and their decay over time. This is done in the form of a histogram bar, with the bar made up of one box for each recommendation. The boxes start off dark and get lighter over time. This combined display shows at a glance whether a resource has been mentioned many times, and whether its popularity is steady or was a fad that burned out over time.

PHOAKS history is separate — in fact, most writers to newsgroups are probably unaware that they are contributing to a history-rich system and, of course, the production of history is by default passive. It is possible that someone who saw the PHOAKS Web site could have then gone and deliberately posted message with URLs in them, hoping to add to or influence the ratings; but this is outside the design of the system. PHOAKS’ passive data collection system avoids the “cast of thousands”

PHOAKS

Reference 37, 77

Reference 36

problem. By hooking into both a large data set and an existing body of practice, they have created a history-rich environment.

HOMR

Reference 69

Another system that did similar things to the previous two systems was the HOMR system developed at the Media Lab. HOMR was a way for people to get recommendations for music they might like to buy or listen to. Unlike the previous two systems, which hooked into existing data streams, HOMR depended on users entering items into its database. Users could add new artists and new albums for others to rate.

HOMR used a statistical matching technique similar to the one described above for GroupLens; however, as the space of users grew large, the simplistic matching techniques began to take too long to compute. HOMR therefore introduced a standard statistical technique of clustering. Users were each given a unique cluster of their 200 nearest neighbors, considering the ratings matrix to be a very high dimensional vector space. This enabled a quick lookup to be made for any user, since the system could avoid pointless comparisons by staying within the user's cluster. The clusters were recomputed nightly, so new data were taken into account on an ongoing basis as history accumulated.

One of the interesting phenomena that grew up around HOMR was the creation of a strong on-line community. Initially, the system did not contain much in the way of public identity. Users could write reviews of groups or albums and sign them if they chose. However, the system builders soon realized that music is something people love to talk about and at that time (early 1994) there were few places on the Web where open chat was supported. Eventually, HOMR was spun off as a company that became Firefly and that was purchased by Microsoft for its personalization and community-building tools.

This emphasis on community led the HOMR/Firefly developers to build more active history tools into their system. For example, later versions supported a home "page" on the system for each user that was linked to reviews written by that person. Users acquired (informal) reputations within the system and built networks of friends. This in turn led to the development of mechanisms for finding people, including quick histories of who the most recent logons were, and who were the most recent visitors to various pages or chat areas in the system.

HOMR's history was distemic in the same way that other recommendation systems connections are. The connection between history and resulting suggestions are tenuous at best and completely opaque at worst. On the other hand, Firefly's quick history mechanisms were much more proxemic, as well as being permeated since they appeared right at the top of each page within the system visited by users. Both systems were highly social, as noted above.

4.3 Structured Navigation Systems

As noted in Chapter 2, one of the important uses of history is to establish authority or authenticity. Typically, this is done in a pedagogical setting. That is, a teacher or expert puts together some kind of course or series of material. Students follow this course because it has the authority of the expert behind it. In hypertext environments, this often takes the form of a guided navigation.

These systems usually do not incorporate interaction history, except possibly in a very limited sense; therefore, we cannot compare them along theoretical lines to Footprints. However, each system we will discuss in this section represents a kind of tool

that is at least superficially similar to the Footprints tools and that is therefore related in some interesting ways.

Zellweger, for example, implemented a tool called Scripted Paths as part of the Xerox Scripted Documents environment. Rather than focusing on an artifact such as a sequence of documents, this tool embedded active computation into the data stream. Typically, a script would consist of a series of documents and locations inside documents, along with actions and timing information. In this sort of environment, a path is generally not followed under user control, but becomes a dynamic presentation of coordinated and linked information. The script is usually played back automatically, although it can also be single-stepped by the end user.

Scripted Documents
Reference 85

Scripted paths were seen by Zellweger and her colleagues as the natural next step in a sequence of increasingly complex path types that begins with sequential paths and progresses through branching and conditional paths. They regarded it as a drawback to have situations in which users must make choices about which branch to follow, where Footprints regards it as a benefit. Interestingly they propose, but do not elaborate on, a form of conditional path in which the path itself could query its own history.

Another advantage of scriptable paths is that they can escape from the constraints of the underlying system's granularity. In Chapter 3 we noted the problems of deciding precisely what constitutes a "page" in the World Wide Web. In Scripted Paths, the elements on the path could be whole pages, sections of pages, or arbitrary points within pages.

See "URL equivalence"
on page 54.

Scripted paths experimented with a number of different visualizations of the paths, including programmatic, network, and list. The system's authors did not attempt to determine if one sort of visualization was best, but instead suggested that programmatic elements within a path could dynamically determine the path's visualization. For example, if the program detected hierarchical structure in the path, an outline-like view could be used. If cross-links between substructures were found, then a network view could be used instead.

Tim Oren and his collaborators took a more direct approach to this kind of application with their Guides system. Their approach used actual characters, role-playing different typical personalities within the application domain, and represented as segments of digitized video. In these segments, the guide characters presented different aspects of a multifaceted guided navigation experience. The experience itself consisted of a series of nodes (episodes) in a branching path (story). Reaching an episode would activate certain of the guides, those that had something to say or show relevant to the current part of the story. Guide characters expressed political and social viewpoints — that of a settler, a Native American, a settler's wife — relating to an historical narrative lesson about America's westward expansion. The Guides system was written for and ran on Apple Macintosh computers and represented the collective authority of a number of domain experts who worked with the software developers.

Guides

Reference 53

The path experience in the Guides systems was linear — a user went from episode to episode, but could get several guides' point of view on each episode. Guides could also suggest which episode to view next, based on the point of view embodied in their personae. The path itself was not directly visualized in the system. Screen space was used for the material in the nodes. The path was strictly a control mechanism.

As the Web has become more popular, it has also grown to contain larger volumes of useful information. To extract the useful parts, people have begun to recreate some of

Walden's Paths

the hypertext-like structured navigation tools on the Web. One typical effort in this direction is Walden's Paths, a system developed by Richard Furuta and his colleagues at Texas A&M University.

Walden's Paths is oriented toward instruction for children in grades K through 12. It includes both path-authoring and -creation tools as well as path viewing tools. A path in Walden's Paths consists of a linked series of Web pages and additional annotations created by the teacher. Annotations may be text or other media such as sound or still images, and they serve to contextualize the pages, provide transitions between pages, and emphasize the parts of the pages that the instructor most wants students to attend to.

Instructors author paths in the Path Authoring Tool, which gives control over a sequence of URLs, a search tool, and an annotation editor. Completed paths are stored in a central database called the Path Server. Students connect to this server, which acts as a proxy similar to the way Footprints works. The Path Server proxy does not create windows for user interaction. Instead, it alters the text of documents before sending them on to the students' browser software.

The proxy inserts a banner at the top of the page that has special "next" and "previous" buttons to allow students to walk the path, as well as numbers indicating the current position and path length. Annotations made by the instructor are shown below this, then the page text. Links embedded in the page are also modified so that if a student follows an off-path link the resulting page is augmented with a button at the top that permits the student to return to the abandoned path.

Walden's Paths notion of path is largely linear, though the authors imagine extending it to a hierarchical structure, with subpaths. It does not have the multifaceted views of Guides, nor does it have the consequent ability to make branching decisions based on coherent points of view. Unlike Footprints, Walden's paths all exist in a flat undifferentiated space. A page may be on many different paths but users cannot see this. Additionally, the lack of a separate tool makes it impossible for students to look ahead. Placing information in context can be done only by virtue of instructor annotations of "flipping" back and forth among pages in the path.

4.4 Non-central authorities

All the systems in the preceding section share the base assumption that a path is a mostly unchanging, top-down designed structure often resulting from a single or central author. However, authority does not have to come from the top down. The last two Web systems we will look at approach the problem differently.

Web Rings

The first, Web Rings, provides a mechanism whereby authors at different locations can link their pages together into a path on a given topic. This path is built in a ring topology, meaning that if someone were to visit it in its entirety she would eventually end up back on the starting page. The topology is maintained within a central database of rings at the Starseed Web site (www.webring.org). When authors add their pages to rings, they do so without knowing where in the topology they will end up. Thus, each page in a Web Ring must stand largely on its own; Ring pages are topically similar but unlikely to be sequentially coherent.

Authors embed, within the HTML of their page, icons that advertise their membership in the ring and that provide links to "next" and "previous" pages. In fact, these links connect back to the Ring server, which looks up the next page in the ring from its database and redirects the user's query to the Web server holding that page. The

Ring server also does some availability and validity checking, so the effect of such a request is usually to get the next available page in the ring, even if that means the ring is different on different traversals.

The ring may also be different because new pages have been added or old ones removed. Although each ring has an “owner” who can deny a page membership in the ring, the owner cannot add pages; only a page’s author may do this, because the author is the one who controls the HTML source of the page. The result is that the content of a path is not derived from a central authority but rather from the distributed authority of all the page creators involved.

Unfortunately, Web Rings are only capable of linear organizations. The author may include icons giving readers the ability to jump ahead or back five or ten documents in the path — a way to ‘skim’ the path — but there is no way to have hierarchical or branching structures. There is also no way to get an overview of the whole Ring or one’s position in it, except that the Ring server always provides the owner’s first page as the starting point for visitors looking to traverse the ring. As with most Web pathing systems, Web Rings exist in a flat space. A page may be on more than one Web Ring, but more commonly there are multiple rings on a given topic. This may happen because there are more people seeking to join a given ring than the owner can accommodate, or because of political, philosophical, or personal differences among the page authors. The Web Ring server keeps a list of active rings, with the rings sorted into general categories, but there is no attempt to make connections among the rings.

Nevertheless, Web Rings represent the only widespread mechanism available today for page authors to work together without being dependent on preplanned structures or central authorities.

Finally, if authority does not derive from the author or authors of the pages, it may be derived from the readers of the pages. Footprints definitely promotes the viewpoint that *vox populi, vox dei* — the voice of the people is the voice of god. In our tools, we treat all transitions made by users as equivalent to links placed in the page by the authors.

Aqui

See “The Web-navigation primitive” on page 54.

IBM created a Web site called *Aqui*, in 1996, that implemented this idea. *Aqui* is based around a database that keeps track of pages input by visitors to the site, and of suggested links. Visitors can suggest new links or vote on the appropriateness of links already in the database. The stated goal of *Aqui* is to find and keep up to date those links that ought to have been included in a page but that were not placed into the page text by the original author.

Aqui appeared to be doing several things at once. Like HOMR/Firefly, it attempted to build something of a community. Users had identifiers within the system and one could see who had proposed a given link and who had proposed the most popular links for a given Web page. Obviously, *Aqui* was also recording a form of history — relationships between Web pages that were not directly linked. The history derived from peoples’ experiences and opinions, a blend of the kinds of history recorded by Footprints and by collaborative recommendation systems. Unfortunately, although the *Aqui* Web site is still functional in some form, it appears that the authors never published any papers about the system, so we cannot know how the system worked internally or what its non-obvious design goals were.

4.5 Architectures

Footprints' implementation architecture — like many Web-based systems such as Walden's Paths — uses a proxy server to perform most of the work. The best overview of this kind of system architecture is the Web-based intermediary framework developed at IBM's Almaden research center.

WBI

Simply stated, the notion of a Web "intermediary" is that it is a program that sits on the HTTP stream between client and server, and performs some computation over, or in conjunction with, that stream. In its simplest form, this computation can take the form of inserting information into the stream so that the resulting HTML is altered. Footprints does this with its annotations; Walden's Paths does this to insert its navigational marker and teacher annotations at the head of each page. Intermediaries can also handle functions outside the data stream contents. For example, a caching intermediary could compare document requests against a local store of documents and then decide either to serve the document from this cache or retrieve it from a remote site.

Reference 2

The intermediary model promoted by Barrett and Maglio is quite general, encompassing most of the work currently done by proxy servers, as well as by client-side programs such as applets and server-side programs such as cgi-bin. To make it easier to work in this complex space, they have also built an architecture and toolkit, called WBI, for Web Browser Intelligence. At heart, it is a programmable proxy server that includes a set of rules and associated priorities and programmatic building blocks, summarized in the table below.

TABLE 1. WBI Building Blocks (from Barrett & Maglio)

Function Type	Input	Output	Action	Example
Request Editor	Request	Request	Modify request	Redirect the request to a new URL
Generator	Request	Document	Produce document that satisfies request, or reject request	Dynamically compute documents
Document Editor	Request and Document	Document	Modify document	Add annotations
Monitor	Request and Document	None	Receive input, perform computation	Gather history information
Autonomous	None	None	Perform background computation	Calculate user browsing statistics

A given intermediary may have multiples of each of these components; for example, multiple request editors could be built to handle different types of requests. The rules part of WBI allows the system builder to control how and when data are sent to the components.

The result of this architecture is a highly flexible, optimized environment for creating a large number of Web-based systems. The authors also propose extending WBI to be able to manipulate document metadata, although it does not currently handle such information. With such extensions in place, WBI could become a preferred platform for implementing a large class of history-rich Web systems.

The beta version of these capabilities (available at the time this dissertation was written) had four personal history-related functions programmed in:

- links on Web pages annotated with “traffic lights” that indicated likely connection speeds;
- keyword search through the text of all pages visited by the WBI user;
- pattern-detection tool that suggested shortcuts, based on paths the user repeatedly followed from a given page;
- change detector that polled “favorite” Web pages to see if they had changed and therefore could be suggested for a revisit.

These tools were not intended to coherently address the question of how interaction history could be embedded in WBI, but rather served to exercise a range of different capabilities within the architecture. In that respect, this effort is similar to Footprints’ visualizations.

In summary, we see that the Footprints tools are related to three classes of implemented systems:

1. History-rich systems, related by virtue of having the same philosophical goals, though most of those systems operate in significantly different domains.
2. Social navigation systems, including recommendation systems. Although Footprints is not intended to provide recommendations per se, its history functions as a form of social navigation. It shares with this class of systems the defining assumption that we can draw useful predictions about current users from analyses of the behavior or input of past users.
3. Path-oriented mechanisms. Paths have been thought of as a useful construct since Vannevar Bush proposed his MEMEX system in 1945, in what has since been widely acknowledged to be the first written conception of what would become hypertext. Prior to the World Wide Web, path mechanisms were built into hypertext tools largely for pedagogical or presentation purposes; these purposes have been recreated within the Web. However, the heterogeneous and distributed nature of the Web allows us to envision new kinds of pathing tools that do not depend on centralized or predetermined structures. Footprints is one such toolset.

4.6 Summary

Reference 9

Finally, Footprints’ implementation architectures places it generally in the class of Web-based intermediaries defined by the WBI work. Although the Footprints proxy server does things not precisely covered in the WBI taxonomy, it is still useful to think about how such intermediaries could support history-rich tools in general.

CHAPTER 5 *Results*

*Caminante, no hay camino
Se hace camino al andar
(Traveller, there is no path,
Paths are made by walking.)
— Antonio Machado*

Reference 46

As noted throughout Chapter 3, Footprints tools are intended to serve as experiments at different points in the space of history-rich systems and were heavily influenced by continuous informal feedback from our users and beta testers. Insofar as the tools proved useful and inspired feedback we may regard them as successful. Two years have given us some insights into how a history-rich system can be used, and this chapter will discuss three of them. Along with the positive insights, we will note some problems with our current implementations.

To measure success in a more formal way and to get a more rigorous sampling of our users' desires, we conducted a survey and a series of experiments. The survey gauged what tools users might want and was conducted before the development of the final system. The experiment measured users' performance on a controlled task both with and without the Footprints tools in an attempt to evaluate their usefulness. In our experiment, we found support for the hypothesis that our tools enable users to get the same amount of work done with significantly less wasted effort. We also discovered that Footprints is more proxemic for — and perceived as more satisfying by — users who are familiar with the problem space, even though they are not familiar with our tools. This was something of a surprise, since our naive assumption had been that our tools would be more helpful to inexperienced users.

Our first insight was introduced in Chapter 3, a description of map components. Experiences — narrative, storefront, in-out, and central message — form the basic elements of interaction history maps. These elements are combined to form the more complex interaction patterns seen in Footprints tools. This leads to two uses. From the point of view of a user, it would be possible to look at the experiences afforded by a history-rich Web site to previous users and decide where to enter the site. A new user might want to start at the beginning of a narrative experience; a more experienced user might skip the introduction and begin directly in the storefront-type area, trusting experience to be a guide. Students could review a central message and, if they did not understand it, could find other ways to approach the topic.

5.1 Insights

See "History Patterns" on page 71.

The converse of this approach is to look at the building blocks from a designer's point of view. Information designers need guidance to organize their presentations, and interaction history can assist with this. In particular, a history-based analysis of users' interactions with a Web site or other information presentation can reveal a mismatch between the intended design and the effect it has. Conventional interaction design gives guidance on the layout of Web pages; history-rich design gives guidance on creating the kinds of user experiences the content creators desire.

See "*Models of Information Navigation*" on page 78.

The second insight was also mentioned in Chapter 3, in the discussion of the multiple senses of a document. The important insight here is that interaction history reminds us that the meaning of a document is more than the sum of its contents. Techniques of keyword extraction and automatic summarization are good, but they ignore the fact that language is a local phenomenon. Words have meaning in part because of who says them, where they are said, and in relationship to other words. As Chalmers and his colleagues argue, the meaning of a document is in part embodied in its relationship to other documents. Interaction history can show the connections people actually make between documents, and which of the information designer-created links are actually used.

Reference 11

A history-rich environment can use this in combination with conventional information retrieval techniques. For example, a text summary could be run over an entire path, rather than one document. Recommendation systems can be extended to suggest relevant trails, rather than single documents. Likewise, designers can organize their presentations to give users multiple trails through information.

5.2 Survey

The survey was given to a group of 21 subjects who had just completed our experimental task without having the benefit of any Footprints tools. The task, which is described in more detail below, involved asking users to browse for information on a possible car purchase, and report whatever alternatives they were able to find within the time limit.

We felt that it would be difficult or impossible to get potential users to describe what tools might be useful, outside the context of any particular problem. We needed both to scope the problem so that we could ask only about tools relative to a certain situation, and also to ask our questions when users were in the proper frame of mind. That is, we believed that if we asked our survey questions at a time when people could relate the questions to an immediately prior experience we were much more likely to get useful results.

See "*Interaction History Framework*" on page 21.

The questionnaire we asked subjects to fill out is reproduced in Appendix B. The survey contained 21 questions on possible forms of interaction history which might be recorded. These forms were generated by taking the 6 parameters from interaction history theory and thinking about how they could be applied to Web browsing. Subjects were asked to give their opinion — on a 1-5 scale where 1 was most and 5 was least — of the usefulness of each possible form of history. The first fifteen were passive forms, such as which Web pages were visited, which links were used, and so on. The last six were active history forms, such as the user's identity. The questions about passive history were particularly concentrated on determining whether users found relative or absolute history measures more useful.

In addition, we had received some feedback from our beta users indicating an interest in seeing a form of "hot" item indication, but confusion over what it might mean for a document to be "hot." Therefore, we asked our subjects to rate three different possi-

ble measures of popularity. On all questions, users' responses could range from 1 (very useful) to 5 (not at all useful).

Table 2 and Table 3 summarize how the individual items rated. Note that three of the top ten items were in the active history section, as can be seen below.

Survey Results

TABLE 2. Passive History Survey Results

Rank	Item Questioned	Mean	Std Dev
1	What pages you visited	1.571	0.597
3	What links you used	1.857	0.727
5	How long you stayed at a page	2.095	1.136
7	Hot items (relative number of visits)	2.333	0.966
8	Hot items (absolute number of visits)	2.381	1.117
9	How long you stayed at a location in a sequence	2.429	1.028
10	How many times you visited a page (relative to other pages)	2.762	1.300
11	Hot items (recency of use)	3.000	1.049
12	Recency of visit to a page	3.143	1.195
13	How many times you visited a page (absolute number)	3.238	1.375
14	How recently have you traversed a link	3.286	1.384
15	How many times did you traverse a link (relative number)	3.429	1.326
16	What pages you didn't visit	3.714	1.007
18	How many times did you traverse a link (absolute number)	3.762	1.044
19	What links you didn't use	3.809	0.981

TABLE 3. Active History Survey Results

Rank	Item Questioned	Mean	Std Dev
2	What you thought of pages you visited	1.762	1.091
4	Why you went to a particular page	2.000	0.894
6	What type of user you are	2.238	1.136
16	What kind of site you connect from	3.714	1.056
20	Who you are (login name/pseudonym)	4.571	0.926
20	What computer you connect from	4.571	0.597

Questions on the survey were phrased in the second person (you) to help subjects think in terms of their personal experience. We wanted subjects to put themselves in the position of being a helper to some unknown other person coming after them — the kind of thing for which interaction history is best, but without our having to explain any theory to them.

The second portion of the survey explained to subjects that the last six forms of information would require them to enter additional data. We debated trying to help users understand that this might be disruptive of their browsing process, but decided the instructions (see Appendix B) were too complicated already and risked confusing users more than helping them understand.

We also asked the subjects to give us their top three choices among the 21 alternatives presented in the survey. The goal of asking this was to guard against the natural

desire of people to want everything when there is no perceived cost or simply to mark down all ideas as good on the survey.

Using a systems of 3 points for a first-place choice, 2 points for a second-place choice, and 1 point for a third-place choice, two items stood out well above the rest. Ten choices that received no votes at all are not listed.

TABLE 4. User Rankings by Points

Item	Points
What you thought of pages you visited	33
What pages you visited	31
Why you went to a particular page	14
How long you stayed at a page	13
How long you stayed at a location in a sequence	8
Hot items (absolute number of visits)	6
Hot items (relative number of visits)	5
How many times you visited a page (relative to other pages)	5
What type of user you are	5
What links you used	4
How many times did you traverse a link (relative number)	2

Since we did not explicitly ask subjects to rank their selections by preference or importance order, it might be argued that a point-based system distorts the intentions of the respondents. Another method would be to simply tally up the number of votes each choice received, without regard to their place in the rankings. The results of this tabulation are shown below.

TABLE 5. User Rankings by Number of Votes

Item	Votes
What you thought of pages you visited	13
What pages you visited	12
How long you stayed at a page	9
Why you went to a particular page	8
How long you stayed at a location in a sequence	5
What type of user you are	4
What links you used	3
Hot items (absolute number of visits)	3
Hot items (relative number of visits)	3
How many times you visited a page (relative to other pages)	2
How many times did you traverse a link (relative number)	1

As can be seen from a comparison of Table 4 and Table 5, there is not much variation in the methods of computing preference. Comparing these tables with the first two rankings of results shows only one large change. The item “What links you used” had the third highest overall mean result, but dropped to seventh or tenth in the rankings, depending on which ranking is used.

This suggests that annotations may be less useful than other Footprints tools, since they directly show which links users followed. The most likely to be useful would be the purposes tool, which attempts to allow users to explain why they visit links, followed by the map and paths tools, which show the nodes used. The results of this survey influenced us to add another tool. In particular, the purposes tool was designed specifically in response to these data; previously we had only planned a comments tool.

We had hoped that the survey would help us resolve the contradictory senses of “hot” which we had gotten in our informal feedback. Unfortunately, though the survey subjects tended overall to rate relative measures as more useful than absolute ones, the difference was not statistically significant. We expect that this reflects a general confusion in the Web user (and Web publisher) community over what actually is meant by the term “hot.” Users probably have a favorite set of sites that apply the term in a particular way and expressed this preference in their responses to the survey.

The next step in our evaluation process was to try to determine the effectiveness of the tools we had built. We have claimed that the interaction history ideas we find in the physical world could not only be captured in a theoretical framework, but then could usefully be applied to solving problems in the digital realm. In our case, the problem we are trying to solve is how to help people reach and comprehend information in which they are interested.

5.3 Experiment

This is a difficult task to measure. In Chapter 3, we showed a number of cases where, by inspection, we could demonstrate how the tools might be used to accomplish these tasks. Our goal now is to measure this in a more rigorous fashion. Will Hill called browsing a “soft landing” task (personal communication). That is, it is hard to measure or detect a failure of this kind of task. If users fail to browse some particular information then they simply browse other information and perhaps end up unsatisfied or more satisfied than they would have been with the original data.

It is particularly hard to measure the usefulness of any particular tool that purports to assist in this form of task. Confounding variables such as level of user experience with the subject matter being browsed and inherent problems in the organization of the underlying information must be taken into account. That is, someone who is better versed in the subject will likely find browsing easier than someone who is not as well versed in the subject matter. Likewise, if the material is well organized it can be much easier to browse than if it is poorly organized. These factors are also not independent. Web sites organized to make navigation easy for the new user may cause frustration in more experienced visitors who want quick routes to information they remember being present.

Such problems are precisely what torpedoed our first attempts at formal experimentation, which involved trying to measure users’ levels of comprehension of a particular topic before and after a browsing session. We found that some testees knew too much about the topic beforehand and so learned nothing in the session, while some knew too little and found the material incomprehensible.

We also considered a test mode in which the test subjects would use one of two different sets of tools for the same tasks. However, our goal is not to analyze the goodness of any particular tool, or to validate any particular visualization of interaction history information. Instead, we wanted an experiment that would help us understand whether history information could be usefully applied to solving our chosen problem.

Task Criteria

See “Alternative Model of Navigation” on page 46.

Since we were unable to test for usefulness in the general case, we set up an experiment in which the usefulness of interaction history could be tested for a specific problem. This required that we be able to frame our task in a way that required elements of both exploration and exploitation and would be measurable along both dimensions. In particular, it is not sufficient simply to require users to find some predetermined answers and measure their speed or efficiency in returning those answers. Although this sort of information searching task is of interest, it does not exercise the system’s capabilities or demonstrate the truth or falsity of our basic hypothesis.

Our chosen task encouraged users to generate alternatives. Our concern is not with the particular alternatives generated, but with their number, and the amount of effort required to generate them. The underlying assumption: a system that aids exploration and exploitation will enable users to generate more alternatives with less effort than operation without such a system. This implies, of course, that users are given a fixed amount of time for the task, which we can control, and that their levels of skill prior to the experiment do not vary too widely. We attempted to achieve this in our experiment by requiring all subjects to be proficient with the Netscape Navigator browser that we were using and by picking a domain familiar to all subjects.

We also wanted a task that would be as realistic as — or close to — subjects’ normal browsing behaviors as possible. We therefore chose not to limit the number, type, or variety of Web sites subjects could visit. Although this risked introducing large inter-subject variations in the experimental browsing experience, we felt it would be unrealistic, and therefore not a valid test of the software, to require subjects to browse only in a prescribed manner or only access a prescribed set of sites. In essence, one cannot say “browse *our* way and we’ll help you” and still claim to be making tools that aid actual Web browsing.

With the task thus designed, we needed to find a “generic” domain in which subjects could operate, and chose the simulated search for a possible new car to purchase. In pretesting the domains of house-purchasing and selection of possible graduate schools by students completing their undergraduate studies, we found that there were too many idiosyncratic features that were not represented in the on-line information. That is, subjects told us they could not make any decision based on the information they could find, no matter which sites they visited. Similarly, the choice of a graduate school was overwhelmingly influenced by intangible features, plus the pool of possible subjects (who had not already made up their minds or who were not already strongly biased) was very limited.

Although ephemeral factors play a significant role in an eventual car purchase, our pretesting found that potential subjects were easily able to articulate criteria that would cause them to be interested in particular vehicles or types of vehicles. We also found that, with the exception of purely personal factors such as a recent purchase by a friend, the criteria were well represented in on-line car information. Additionally, car purchasing is a decision that is considered by people of widely varying ages, backgrounds, and economic situations. Where the first two domains restricted us to relatively narrow pools of potential subjects, the car domain led us to test with subjects who ranged from their late teens to their early fifties, and from low to upper-middle class income status.

The Task

The task as finally run gave subjects 20 minutes’ browsing on the Web to find cars that they might be interested in buying. Our pretesting had found that 20 minutes was long enough for most subjects to feel that they had accomplished something, but not long enough that they would start to become bored. Few of our subjects (5%)

declared that they were finished before time was up. We did not provide subjects with a clock or timer, but we gave them a warning five minutes before the time was up. This was done so that they would focus on the task and not be overly concerned with the amount of time they were taking.

Subjects were asked to imagine that they had approximately \$20,000 to spend on the purchase of a car. This amount was chosen for two reasons. First, it seemed like an amount that would be understandable by a majority of subjects and would offer a large pool of car types and manufacturers from which to choose. It did rule out a few of the high-end models, but we felt this was not a significant drawback. Secondly, our pre-test showed that \$20,000 was a common breakpoint for the organization of models into categories in on-line presentations by car Web sites. In particular, a number of these sites offered visitors the option of choosing cars in the range of \$15,000-\$20,000 or \$20,000-\$25,000. Thus we hoped that by specifying an amount approximately on this border, subjects would be encouraged to explore more than if we had picked an amount which fell neatly into preexisting categories. At the same time, this constraint reduced significantly the set of options which users might feel compelled to examine. Usually a manufacturer will not offer more than two or three basic models in any given category.

Beyond the constraints of time and approximate price, subjects were encouraged to bring and use their own preexisting ideas, attitudes, methods, and preferences. Subjects were told that if they were only interested in sports cars or station wagons, or whatever, that this was fine, and they were not required to examine vehicles that would not ordinarily interest them. Additionally, subjects were told that they could use whatever Web search and navigation strategies, sites, and techniques that they would normally use.

Our testing strategy was to have subjects perform the same task in either an unaided or with-Footprints condition. For the testing in condition two, we used a fixed database, which was the same for all users. Since we were attempting to make our task as close as possible to what users would see in a “real” situation and not introduce our own biases too heavily into the data set, we used data from the first group of users as the history available to the second and third groups. It would have been easy to hand-create a fake database with history information that led users through the task in a neat, orderly fashion or that was comprehensive enough to cover most of the car-related sites we would expect users to visit. The resulting database was biased, of course, but biased in ways reflecting the idiosyncrasies and behaviors of an actual community of users.

Testing Strategy

Footprints attempts to enhance the experience of browsing; therefore, we wanted to have subjective measures to capture peoples’ sense of the experience. These senses are not replacements for objective measures, of course, but they are important in getting an overall idea of whether Footprints contributed to, or detracted from, the browsing experience, and how users felt about what they had done. Our objective measures were the number of alternatives generated — i.e. the number of possible car makes and models found — and the number of pages visited in order to generate these alternatives. The goal of having these dual metrics was to gauge the ability of subjects to exploit the information and their efficiency in doing so. The underlying assumption is that there are likely to be a number of intervening pages and a large number of false starts built into the space of Web pages that subjects might visit. Our intuition was that the history structures and visualizations provided by Footprints would allow people to avoid these false starts and cut down on the number of intervening pages. Thus, we hoped to see the number of alternatives generated increase and the number of pages visited decrease.

Testing Metrics

See “Post-Test Questionnaire” on page 137.

Our subjective measures aimed to answer three basic questions: How well did the exploration go? How easy was it to do the task? How good or bad was the organization? Each question was asked in two different ways, to separate possible different effects. For the first question, we wanted to measure how well subjects felt they had covered the space of possibilities — a metric as closely related to exploration as we could get in an unconstrained browsing task. We asked subjects to tell us how satisfied they were with the car choices they had written down (Q1). We also asked them how satisfied they were with the experience of browsing they had during the experiment (Q2). Our goal in splitting out these two questions was to see if there was an effect of Footprints on either the perceived quality of the results or the quality of the experience.

We next asked about ease of use. Specifically, we asked subjects how easy it was to find relevant information (Q3) and to understand what they saw (Q4). Obviously our hope was that using Footprints would result in improvements along both dimensions. In fact, it would be remarkable if there was significant improvement, given that subjects were completely unfamiliar with Footprints prior to the experiment. Given the level of complexity of the tools, it would be surprising if subjects found it easy to use at first. Ideally, we would have been able to run this test on a group of subjects who were familiar enough with Footprints not to be considered new users, but the only people local to us who have used the system extensively are friends and research group members who have too much knowledge of what the task is to be counted on for unbiased results.

Finally, we asked about the perceived organization of the information in two ways. The first (Q5) asked how well-structured the subjects thought what they saw was; the second (Q6) asked how distracted they were. We hoped that Footprints would improve the first metric — that is, help people perceive structure in what they saw while at the same time not increasing the last metric. As we noted in Chapter 2, interaction history systems must take care that the manipulation of history does not override peoples’ ability to engage with their chosen tasks.

See “Property 4” on page 25.

TABLE 6. Comparison of Basic Performance

	Q1	Q2	Q3	Q4	Q5	Q6	#alt	#pages
Expt 1 mean	2.35	2.5	2.15	1.6	2.35	2.5	3.947	24.8
Expt 1 stdev	0.745	0.761	0.988	0.681	0.933	1.1	2.222	7.4
Expt 2 mean	2.55	2.5	2.25	1.95	2.45	2.25	3.05	18.75
Expt 2 stdev	0.945	0.889	1.07	0.945	1.146	0.91	2.09	9.43

In Table 6, “Expt 1” refers to the first experimental condition, unaided browsing. “Expt 2” refers to the second experimental condition, aided by Footprints’ tools. Each column corresponds to one of the measured items, either subjective or objective, outlined above. Remember that 1 is best in each of the responses, so the lower the number, the more favorable the subjects’ reactions.

In this comparison, only one result was significantly different, that of the number of pages visited. The second group, using Footprints, was able to generate the same number of alternatives but required significantly less effort to do so. The natural question that derives from this is Why? Our hypotheses above suggest an explanation. We also analyzed the data and performed a double-check experiment.

The data were analyzed by grouping the subjects into categories based on their responses to the first part of the questionnaire. That is, we checked for significant variation based on subjects' sex, native language, prior Web and real-life shopping experience, and Web expertise. We found some variations within group, and one across groups. Neither method of analysis showed any statistically significant variation in the objective measures.

Within the first group of subjects (labeled as "Expt 1" in the tables above), a gender-based analysis showed that women (mean 3, sd 0.894) found the browsing experience significantly ($p < 0.05$) less well-structured than men (mean 2.2, sd 0.941). In many cases, we had too few subjects to perform a statistically significant analysis. For example, the first group of subjects included only three people whose native language was not English so nothing could be said about differences in that parameter.

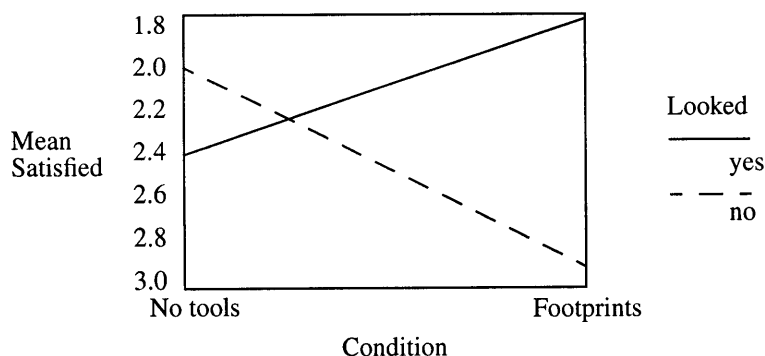
The second group of subjects produced more intragroup variation. Nonnative speakers (there were 6) were significantly more satisfied with their experience ($p < 0.05$, mean of 1.83 versus 2.89). There was also a significant difference between the male and female subjects in group 2. On questions 2, 4, 5 and 6, the ten female subjects in group 2 reported significantly worse results. This appears to be an instance of a well-known phenomenon in which women (or girls) will self-report poorer performance even when the objective measures show no difference in performance. In fact, there was no significant difference between the males and females in objective measures of performance.

Previous car-shopping experience did not have an effect on the results; however, the fact that a subject had looked for car information on the Web before was significant. The 9 subjects who had found it significantly easier to understand what they were seeing ($p < .05$, mean 1.89 vs. 3.00). Level of previous Web experience also produced one significant difference in the subjective measures. More experienced users reported significantly better structure in what they say ($p < 0.05$, mean 1.67 vs. 2.73). These findings are interesting because they indicate that Footprints tools are not compensating for the confusion felt by inexperienced users. We had hoped to design tools that would help new users; instead, Footprints tools turn out to be more useful for experienced users.

Effect of previous car information browsing

This conclusion is reinforced by our single intergroup finding. We compared subjects from different groups across each of the self-reporting features and only one case showed an interaction effect. This effect is illustrated in Figure 41 which shows that, compared across experimental conditions, those who had looked for car information on the Web in the past were much more satisfied with using the Footprints tools.

FIGURE 41. Cross-condition Comparison of Satisfaction



Remember that in answer to all questions in our experiment, 1 is the best possible answer; therefore, lower numbers are placed higher on the graph. This chart shows that in the first condition (no tools), subjects who had previously looked for car information on the web (the solid line) were not more satisfied than those who had not looked, while in the Footprints condition, subjects with this kind of experience were significantly more satisfied. Note that the difference between the two groups in the “no tools” condition is not statistically significant, so we cannot say for certain that the more experienced subjects were less satisfied in the “no tools” condition, though the chart may give that appearance. There was no interaction effect for previously having searched for car information and native language.

See “Models of Information Navigation” on page 78.

Our post-hoc explanation for this phenomenon is that we have inadvertently demonstrated an effect which we had previously noted. In Chapter 3, we discussed the effect of Footprint’s visualizations in giving users access to the cognitive models of information organization created by past users. Simply put, the way people interact with an information space should be shaped by how they *expect* the information to be organized. This may or may not match up with how the information is actually organized.

In fact, on the Web it rarely matches up. If it did, users would have much less trouble navigating than they do. However, when we allow people to see patterns, it is possible that such patterns can be recognized. This is, I believe, what is happening here. Subjects who have experience with car information on the Web have formed models of how the space is organized. With Footprints, they can see others’ models and recognize similarities between their own models and what they see. This leads to their feeling more satisfied because they are less lost and less bewildered. Our post-test informal chats with subjects lend credence to this interpretation. Several subjects mentioned that they “recognized things” in the Footprints displays and homed in on what was familiar to them. It was the subjective impression of the test-givers that these subjects seemed the most pleased when we talked to them afterward.

For the second group of subjects, we also asked three questions about their preferred style of navigation. In specific, we wondered whether a map, written directions, or spoken directions would be preferred by our subjects and whether these pre-test preferences would correlate with any differences in their experimental performance or experience in doing the task. Unfortunately, these questions generally did not differentiate our subjects. Only the second question (written versus spoken) produced a significant split, with 9 subjects preferring spoken and 11 preferring written. This differentiation did not correlate with any new differences on the subjective or objective measures. Our subjects overwhelmingly preferred the map over the written directions in both other case (19 to 1, and 17 to 3). This probably reflects the biases of a subject population drawn from in and around the MIT community. This reminds us that although our subject population was fairly diverse in gender, age and social status, it is probably not a representative sample of the entire Web-using population.

Failure to Scale Up

As noted above, the database used for the experiment was formed by collecting the work of the first group of subjects. Although we believed that our database was useful and appropriately constructed, the scalability of the Footprints approach was still uncertain. To double-check our beliefs, we ran a smaller group of ten subjects through a repeat of the second condition. However, this group used a different history database — with all the data from the first group of subjects and all the data from the second group. This database was slightly more than twice the size of the database we originally used.

As Table 7 shows, the verification group of subjects did not score statistically significantly different than the previous group on either the objective or the subjective measures. However, the trend is definitely toward worse results.

TABLE 7. Comparison of Large and Small Database Effects

	Q1	Q2	Q3	Q4	Q5	Q6	#alt	#pages
large DB mean	2.9	2.8	2.4	2.1	2.7	2.3	3.1	22.8
large DB stdev	1.197	0.789	1.35	0.994	1.16	1.42	3	18.8
small DB mean	2.55	2.5	2.25	1.95	2.45	2.25	3.05	18.75
small DB stdev	0.945	0.889	1.07	0.945	1.146	0.91	2.09	9.43

One way to deal with this scaling problem would be to have automatic filtering available for the more complex representations, such as the map; this would be effective at handling scaling over a short period of time, as happened in this experiment when many people visited the same small set of sites. Another way to help the tools scale would be to institute a fade function allowing older data to be removed from the system.

A more sophisticated method would be to implement a disconnected form of history as an information “scent” in Pirolli’s sense. The idea would be to use history in such a way that a person at any point in the representation could pick up traces of past activity at remote locations. Ideally, this would involve something like a real-time model of the user’s current interests (for example, what their currently selected purpose is) and a visual relationship between the current model and information some distance away.

Reference 56

In thinking about these results it seems likely that Footprints tools require more capability for filtering if they are to scale to handle very complex spaces. Currently there are no automatic mechanisms to discard or hide history information. There is no fade function in Footprints other than that which removes data associated with Web pages that are no longer valid. Given the rapid evolution of the Web, this had been sufficient for normal operation by our dispersed user community. However, placing a large number of people into a relatively concentrated area of the Web clearly seems to require additional fading or filtering capabilities.

See “Fade function components” on page 24.

Information scent can also help with another possible negative effect, which we sometimes call the “lemmings problem.” The design of Footprints is such that popularity is represented very directly in the interface and can be seen over a large amount of information such as all the documents in a map view, or all trails in a path view.

The Lemmings Problem

Our attitude and design philosophy throughout this project has been *vox populi, vox dei*. If everyone wants to go to the same Web site, or wants to follow the same trail, we do not feel that we should discourage this behavior. However, it can lead to detrimental effects. If people blindly follow the most popular path and that path leads to something that ends up being felt by most users to be a waste of time, that path will still get reinforced and will, in the Footprints visualizations, appear more popular as it gets more traffic. Users could leave a comment at the end document or the trail segment leading up to it, or could even leave a warning at the head of the trail indicating that the trail was a waste of time. However, these warnings are not directly viewable, whereas the popularity of a document or trail is viewable and this combination can

lead to a positive feedback loop of more and more people following an ultimately worthless trail.

See “Footprints Paths Viewer” on page 76.

Information scent can help this by providing a more detailed idea of the end result at the head of the path. This can be seen to some extent in the current implementation of the Paths tool, with the ability to press the “Show Titles” button and see the titles of all nodes in the path. However, it would be valuable to be able to propagate the “scent” of opinion registered through comment or evaluation closer to the heads of paths.

Some protection against the formation of instant “crowds” or lemming trails is built into Footprints. First, there is a moderating effect in the way the database updates. Passive history, which leads to increases in popularity rankings, is only added to the database in batch mode, when the overnight update is run. Active history, such as comments, is made available to all users immediately. Thus a negative (or positive) evaluation can potentially be seen much sooner than a change in popularity. Second, popularity is not measured on an absolute scale. A trail may show that it is “Heavily Used,” but it cannot show that it is the “Number 1” or “most popular” trail in the system.

Reference 50

Even without explicitly asking users for rankings (or using comments) it is still possible to infer the usefulness of a path or parts of a path by watching how users move along it. In particular, we expect that dwell time will be a valuable metric for estimating usefulness. It is easy to imagine a modified Path viewer that visualizes the usefulness of paths and parts of paths based on the mean dwell time users.

Use of Footprints Tools in the Experiment

Another question we answered is just how much the subjects in our second experimental condition used the tools. Remember that the instructions they received explicitly reminded them that use of the tools was not mandatory. Additionally, being novices with the system they no doubt missed instances where they might have made use of the tools, and they may have been intimidated by a system with which they were unfamiliar. Therefore, we performed a post-hoc analysis of the log files generated by 19 of the subjects who were using our tools.¹

Footprints contains instrumentation such that it records all navigation actions. That is, whenever the user loads a new page the system writes to a log file the page that was loaded, and the method used to load it. Thus we can analyze the ways in which user navigation was performed, though we cannot record non-selection actions within the tools such as moving the map around or turning a node’s title on or off.

Table 8 shows that about 45% of the users’ navigation actions were taken through the Footprints tools.

TABLE 8. Subjects’ Use of Footprints Tools

Action	Percentage
Map	4.41%
Path	40.77%
Annotation	0.41%
non-Footprints	54.38%

1. One subjects’ log files were corrupted by a file system error and could not be analyzed.

Overwhelmingly, subjects chose to use the Paths tool when they chose to use any tool at all. Our post-test chats with subjects indicated that they probably felt more comfortable with the form of this tool since it did remind them of outline browsers with which they had experience.

When we factored our subjects into groups we found a surprising result. We compared the results obtained by those who used each tool more than average against the results obtained by those who used the tools less than average. In particular, we found that subjects who used the path tool more than average did not do significantly better or report significantly better results than those who used it less than average. However, those who used the *map* tool more than average did better, in two ways.

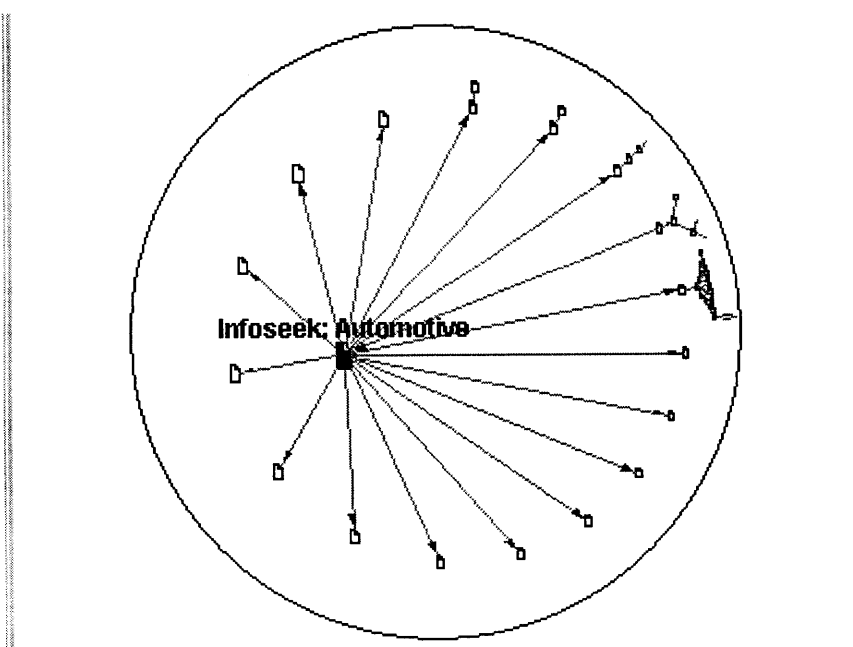
Above-average use of the map tool contributed to both fewer page views needed and a significantly easier-to-understand experience for subjects (Q4 on the post-test). Below-average-use subjects viewed a mean of 24.63 +/- 11.81 pages, while above-average subjects viewed 15.45 +/- 4.5 pages. The other significant ($p < 0.05$) difference was in subjects' self-reported ease of understanding what they saw. The above-average group gave a mean 1.45 rating (0.52 sd) versus a reported mean of 2.5 (1.07 sd) for the below-average subject group.

We feel that this strongly supports our hypothesis about the effect of previous viewing of car information on the Web. Approximately 50% of each subgroup (4 of 8 below-average map users and 5 of 11 above-average map users) had looked for car information on the Web prior to the experiment. The difference in performance was made, we hypothesize, by the map tool because it provided a pattern-recognition opportunity that the simpler representation of the path tool did not afford. Unfortunately, we were not able to videotape our subjects or interview them afterward in depth in order to support this hypothesis.

See "Effect of previous car information browsing" on page 115.

FIGURE 42. Typical Experiment Map Cluster

What Subjects Saw



We did look at the data presented to subjects to see if anything was immediately obvious about the particular interaction history data set with which they had been presented. Figure 42 above shows a typical cluster from the experimental maps. This

See "History Patterns" on page 71.

cluster centers around Infoseek's automotive page and shows a very typical in-out pattern. Most of the moves off this page are only one link deep. A few are two or three and only one led to any fruitful results.

FIGURE 43. Fruits of Infoseek

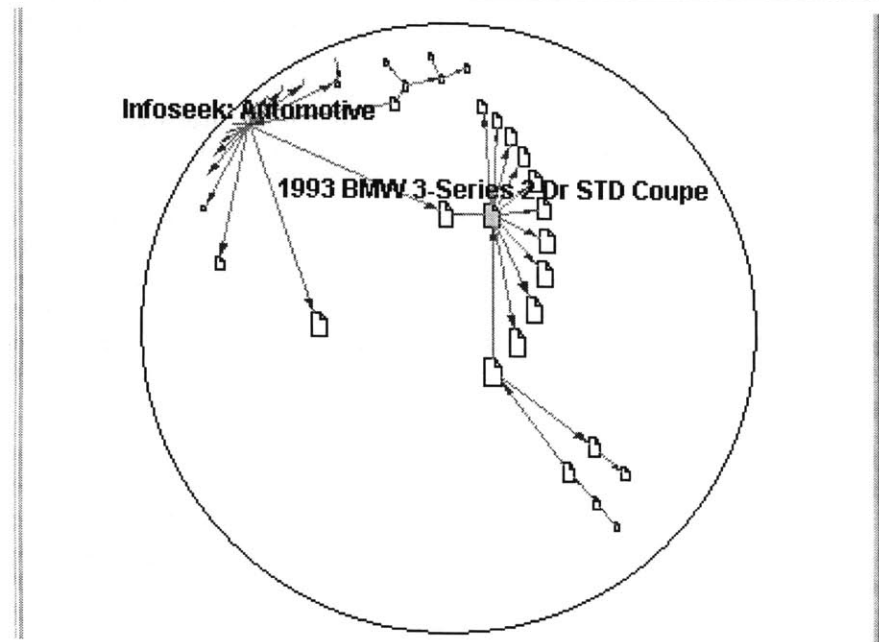


Figure 43 above shows the one useful cluster generated from Infoseek by our subjects. It also shows an interesting example of a case where the in-out pattern was not particularly negative, unlike in Figure 42. In this case, the pages leading away from the page about the BMW give further information about the car. We might argue against a Web site design that requires users to load so many separate pages to answer their queries, but at least there is information to be had.

The most popular site used by subjects was Edmund's², as shown in Figure 44. Four of the most popular documents used by our subjects are in this cluster. Edmund's navigation was largely in-out, but with some cross-navigation, probably created by subjects who were already familiar with the Edmund's site. Edmund's site includes a series of buyer's guides, each of which is usually no more than two levels deep. In this respect, it may also be seen as a central message. That is, users return to the central list of buyer's guides after visiting each guide³.

The paths going through the Edmund's site are also typical of the sort seen by our subjects. As shown in Figure 45, the path starting at the Edmund's guide has average weight, meaning it is used as much as most other paths in the database. Interestingly, the paths which lead to Edmund's from Yahoo are more lightly used. This reinforces our contention that search engines are not as useful as prior knowledge.

2. The Web version of a popular print magazine devoted to cars and car evaluations.

3. No doubt this is some of the most valuable (and expensive) advertising space on their Web site.

FIGURE 44. Edmund's Map Cluster

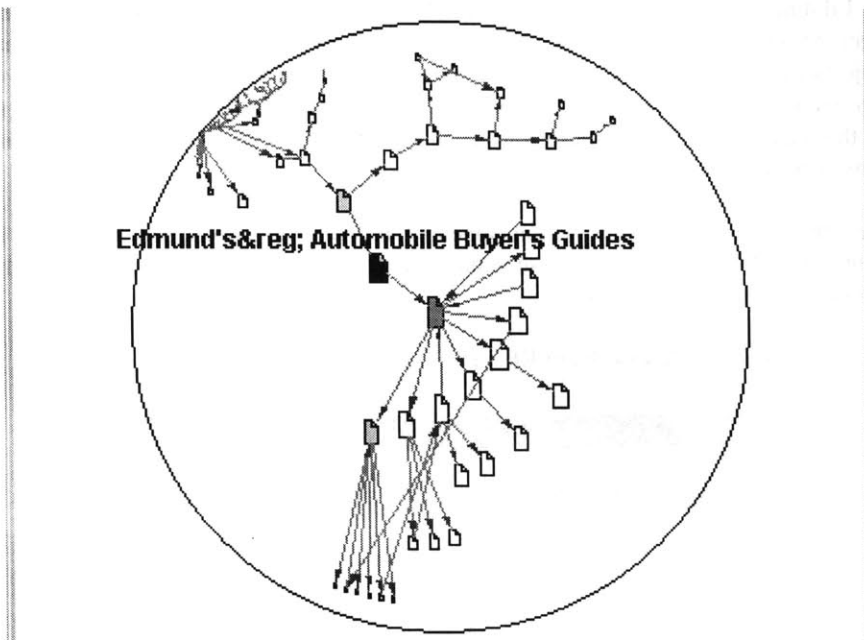
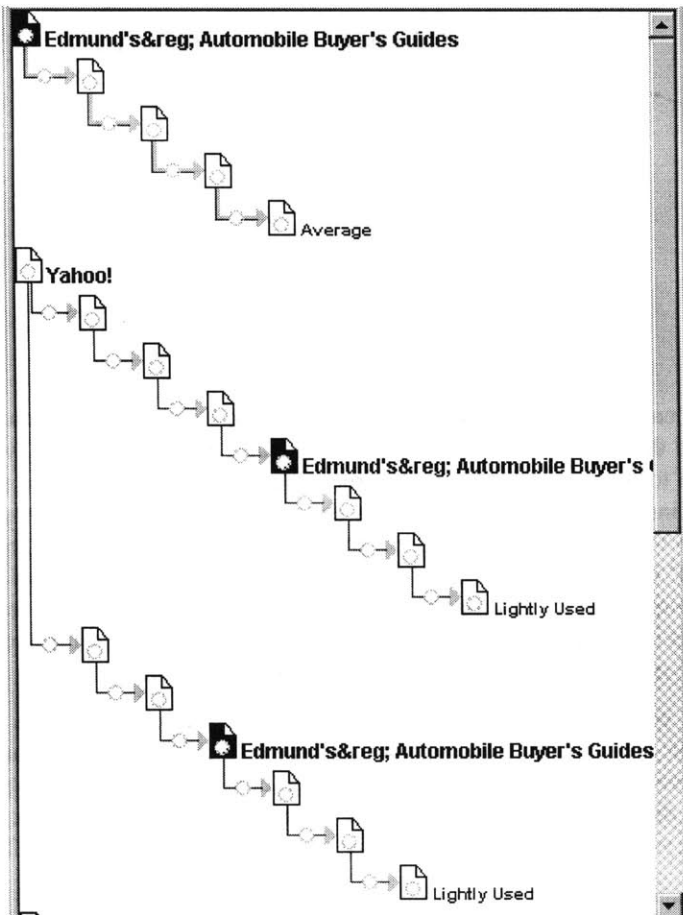


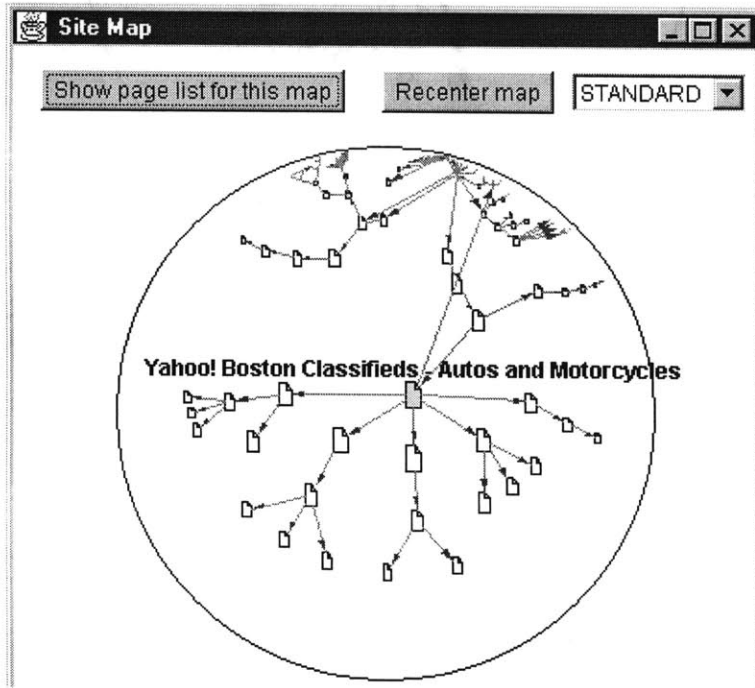
FIGURE 45. Edmund's Paths



It is interesting that the paths which come from Yahoo are the same length after reaching Edmund's as are the paths that start there. This suggests that the intervening pages do not add any particular value. This reinforces our main finding above that subjects using Footprints were able to accomplish the same task while requiring fewer page views. It is not unreasonable to suppose that users with Footprints were able to see that people starting at generic sites such as Yahoo had ended up at Edmund's and were therefore able to omit the intervening pages.

One other interesting phenomenon appeared in our inspections, and is shown in Figure 46. Here we find a definite central-message experience oriented around the Yahoo! Boston Classified ads.

FIGURE 46. The Boston-local Effect



This is likely caused by the fact that most of our subjects are at least somewhat local to Boston and the experiment was run from a computer close to Boston. Two explanations come to mind, and both may be true. First, subjects might have taken the experimental instructions somewhat literally and searched for cars in the Boston area, bringing them to Yahoo's classified ads for the city. Second, the Yahoo! Web site may have detected⁴ that the person asking for car information was in the Boston area, and redirected the query to the Boston-relevant classified ads. This situation reminds us that Footprints is not the only tool active on the Web, and raises concerns about possible 'spoofing' of history data, a topic which we discuss in the next chapter.

Problem with the Task

In discussing weaknesses of our experiment, we must also acknowledge a potential problem in our chosen task. Although we attempted to craft the task as carefully as we could to match our requirements of containing both exploration and exploitation elements, we did not fully succeed. If I ask you what kinds of car you are thinking about buying, you will probably give me a list of two or three alternatives. This is, not

4. Based on the IP number of the querying computer.

surprisingly, what most of our subjects did. Ideally our task should have been more open-ended and, thereby, encouraged people to generate more alternatives.

Of course, there is also a question of how closely any laboratory task can mimic the kinds of browsing people do on their own. In particular, it is probably rare for people to focus exclusively on one thing for as long as 20 minutes. Although Footprints was developed as a set of experiments and not a commercial toolset, I had hoped to be able to make a system which would have an impact on cognition in the wild.

In these experiments, we demonstrated both an expected and an unexpected result of our tools. We hypothesized that having interaction history information available would enable users to accomplish a task with less effort, and demonstrated that this was the case. We found that subjects, even though unfamiliar with Footprints, could still recognize and take advantage of patterns in the history displays. Our chosen task tried to include elements of both exploration and exploitation while still mimicking user's normal Web browsing behaviors as closely as possible. Given our results, we can say that Footprints is definitely an aid to exploration, in that it helps people move more efficiently and satisfactorily through the space. However, we cannot say that exploitation is helped by our system, since users with the tools returned the same number of alternatives as users without.

5.4 Summary

It is good, at certain hours of the day and night, to look closely at the world of objects at rest. Wheels that have crossed long, dusty distances with their mineral and vegetable burdens, sacks from the coal bins, barrels, and baskets, handles and hafts for the carpenter's tool chest. From them flow the contacts of man with the earth, like a text for all troubled lyricists. The used surfaces of things, the wear that hands give to things, the air, tragic at times, pathetic at others, of such things, all lend a curious attractiveness to the reality of the world that should not be underprized.

In them one sees the confused impurity of the human condition, the massing of things, the use and disuse of substances, footprints and fingerprints, the abiding presence of the human engulfing artifacts, inside and out.

Pablo Neruda

Reference 51, p xxxi

Fundamental to any dissertation is the status of its central thesis. Given the results and insights discussed in the preceding chapter, we can say with some assurance that we have proven our central thesis — that interaction history can be usefully applied in the digital realm.

We have shown how interaction history can be applied to two specific problems. For information foragers, we have shown how it can be helpful in finding desired information and how it can be used to place information in an activity-centric context. For information designers, we have suggested how interaction history might be blended into an interface design and what design concerns should be in creating a history-rich system. We have also suggested how interaction history tools can give insights into the models and perceptions users have. Such insights can be helpful in the iterative redesign of things such as databases, Web sites, or even physical spaces.

We began by describing a class of physical-world phenomena. We developed a framework for describing these phenomena and showing how they play a role in:

- skilled reflective practice,
- everyday purposeful conversations, and
- the social construction that is our memories.

Artifacts and their interaction histories tell us stories that give guidance and help us solve problems in the present. These stories form an integral part of skilled practice

in the real world — cognition in the wild. In this practice we make exploratory changes and read back the results of the changes we and others have made. We have shown how a group of people, distributed in time and space, can use a technology like interaction history to collectively form and function as an integrated memory system.

Our theoretical framework operates predictively to provide guidelines and identify important issues for systems builders, as well as retrospectively to help us analyze existing systems. We see that personal history systems are related, as are ratings systems and social navigation systems. The framework also gives us a language for talking about these sorts of systems and describes a space within which all such systems should be able to be placed.

See “The “Book on Java” Story” on page 15.

Lastly, the framework identifies important cognitive properties of users of interaction history systems and suggests a process-oriented view of their design and use. That is, we suggest that sequences and connections are as important as the contents or identities of artifacts. As we first described in our Java book story in Chapter 1, an essential part of the meaning and understanding of an artifact comes from its context of use.

6.1 Improvements

No such frameworks are perfect, of course, and we have repeatedly identified the Footprints tools as experiments that explore interesting points in the theory space. Experiments never completely succeed; often their failures are more instructive than their successes. They give us insights into ways that we can improve the situation in which we experiment. In the case of this dissertation, we identify three major areas for improvement. The first is in the theoretical framework, the second is in the particular tools built and the third is in exploring areas not covered by this dissertation but that are related in interesting ways, particularly the area of identity.

Theory Improvements

The primary improvement that could be made in the theoretical area would be an articulation of the interrelationships among the properties of the framework. As noted in Chapter 2, the theory elements are called “properties” rather than “dimensions” because they are not precisely orthogonal. Although we talk about them describing a space of history-rich systems, that space is not precisely defined. Some of this has to do with the fact that humans do not think in precise orthogonal ways — when we think about one property it inevitably affects others. Thus, there is a relationship between, for example, the proxemic/distemic property and the active/passive property. Both relate to the user’s mental state and it may be that passive history is more proxemic in that it does not take the user out of normal task thoughts. On the other hand, a distemic system can be passive, as rules we have memorized (such as how to drive on the highways) can become so familiar we execute them without much conscious thought. Certainly we do not think of them as restrictive until we want to violate them (perhaps by speeding). Similarly, we may find permeated digital history more proxemic because it more closely mimics the forms of history we use in the physical world.

As the theoretical framework is applied to system design, other interrelationships will need to be taken into account that are specific to the domain in which the system is being built. For example, the domain of application will provide constraints on the kind of information which is available, due to legal or practical limits. In the physical world, if you walk into my store I can tell at a glance your apparent age, sex, height and weight. Peoples’ appearances and chosen modes of dress give significant clues as to their areas of expertise, origin, disposable income, and attitudes. If your browser software brings you to my Web storefront, all these cues are missing. A history-rich system that needed this kind of information could not capture it in a passive fashion

— an active mechanism such as a site registration or completion of a questionnaire would be required.

Instatiation of more history-rich systems will also help us come up with more concrete guidelines for interface designers in applying the theory. For now, we have only the suggested order in which to make the decisions that the theory calls for. Understanding the linkages between properties will help designers anticipate consequences of their decisions. Experience will also help determine things like what are appropriate fade functions — something for which implementers of these systems currently have only a vague intuitive feel.

Another area where the theory could be enhanced would be in enumerating links between design of history-rich systems and other theories of user interface design. Methods well-known in the user interface field, such as task breakdown and contextual analysis, can supply information relevant to designers in deciding what history information can be captured, how to make systems more proxemic, how to implement the “interface physics” that Bederson talks about.

Instantiation of interaction history theory in an application would also be helped by an examination of two properties of history-rich representations that Footprints mostly lacks: *information scent* and *carrying capacity*. Scent is the notion that traces relevant to a distant object may be found locally. For example, the Footprints path tool allows users to see the title of a terminal node from any location on the path. Similarly, the Footprints Web site gives keywords and usage information for a path before users begin to follow it.

This technique could also be applied to other areas. In particular, the ability to leave an information scent for comments, particularly if the scent could carry a positive or negative connotation, could be invaluable. The “lemmings” problem — many people following a popular path that leads nowhere, or nowhere of value — would be vastly helped by this kind of scent propagation. In addition, scents could be combined so that the overall Path could acquire a scent composed from all the comments applied anywhere on it.

Adding explicit positive/negative value to the feedback would, however, require more guards against spoofing, as discussed below. The value of a remote opinion scent is that it could help people decide whether they wanted to follow a particular trail. As noted in Chapter 5, subjects who used the Path tool more than average did not perform better than those who used it less than average. This suggests that the path visualization should be enhanced, perhaps with an information scent, in order to make it more effective.

As noted in Chapter 2, the carrying capacity of a representation is a measure of the number of bits (amount of information) that can be recovered from it. Most data representations compress or “lose” information in some way — for whatever reason they cannot or do not show all the data which is available. Ideally, they would hide only the unimportant or redundant bits, leaving the most salient and most critical bits highly visible. In reality, of course, this idea is only approached, never achieved. In part this is because of the limits of our technologies — screens, fonts, etc. — an in part it is because we cannot ever really know in advance precisely what are going to be the important bits for any particular user of the visualization. In information foraging tasks, where serendipity plays a significant role, this is particularly true. In order to promote serendipity we cannot simply give the user what is desired; we must offer

Reference 7

See “Pad++” on page 90.

*Representation
Improvements*

*See “Carrying Capacity”
on page 26.*

additional information in the hope that it will spark the kind of “aha!” which characterizes serendipity.

Also, we must weigh carrying capacity concerns against or with other kind of concerns, which might lead us to reduce the capacity of the representation. For example, the Footprints map has what appears to be a low carrying capacity. It takes up a significant amount of screen real estate, and seemingly wastes this space. Yet our experiment showed that subjects who used the map more than average were significantly more satisfied. We hypothesized that the map provided an opportunity for an “aha” moment of recognition — a kind of gestalt perception. It is not clear that a denser, more data-rich, representation would have provided the same opportunity for recognition. A representation with higher carrying capacity might have been more distemic by requiring users to learn additional rules to decipher the denser depiction.

*Measurement
Improvements*

Although our experiments gave us some interesting results, they are still limited and could be improved upon. In particular, what they cannot tell us is what the effects will be of using a history-rich toolset over an extended period of time. Five minutes of pre-task training may have familiarized subjects with the basic capabilities of the tools, but they cannot give us insights into the kinds of behaviors which would be exhibited by users who had become experts with Footprints.

In particular, our user base was subjected to a constantly changing set of tools, tool capabilities and functions, and even interface designs. In the 18 months that Footprints was available to be used outside our Lab, three major versions were released — each adding a new tool or major capability. Version 1 contained only maps; version 2 added paths. Comments and purposes were not available until version 3 of the system. In addition, interface design changes were made to enhance learnability of the tools as well as to incorporate user feedback and requested features. This most likely interfered with users’ ability to build up long-term experience with the toolset. It would be interesting to measure the effects of a stable set of functions and features over a longer period of use. We might look to see which tools are used more, and how the effectiveness of using the particular tools changes over time.

The change in interface tools was also accompanied by changes to the back end and to the Java classes used to store the data. The result was that some of the data produced by early users was invalidated. A more stable system would have a more stable representation that would allow accumulation of larger amounts of data and would permit users see the database evolve in response to their activities over time.

It would be fruitful to analyze across a longer period of time the data produced by users of Footprints. Obviously issues such as scaling, filtering and the proper fade function(s) could be explored. We could also do a longer-term analysis of the user experience types we began to identify in Chapter 3. Relative frequencies of occurrence of each of the pattern types could be tracked. These could be compared against variables such as document types, classifications of Web sites (news, sports, entertainment, personal, etc.), and different sort of users. We hypothesize that these experience types could be useful components for an information designer contemplating the effects of an information presentation on users. It would be interesting to put this sort of analysis tool in the hands of experienced Web designers and see if their use of the tool in an information design/redesign cycle allows them to significantly improve the usability of their sites.

In particular, it would be useful to apply a set of history-rich navigation tools to some kind of intranet — that is, a network like the World Wide Web, but restricted to use by

employed, clients or customers of a particular company or organization. Intranets are usually set up as repositories of material which the organizer feels will be particularly helpful to solving problems. Often an intranet is specifically established to give people access to ideas or solutions from past experiences. For example, an intranet might contain answers to all questions asked by customers who called a company's product service center. These prior solutions are intended to be reused to help with new problems or to be given to new customers who call with the same problems. With the user group thus constrained and the material on the intranet both more focussed and more oriented towards foraging for ideas and answer, a Footprints-style toolset could be particularly valuable. This is, in essence, the kind data set we used in our experiment to help the second group of subjects.

The notion of a constrained user group suggest that another avenue for further research would be history-rich personal systems. The Footprints tools are not just accidentally social — they deliberately rule out all notion of individual identity. This decision was made in part for reasons of originality; as noted in Chapter 4, extensive work as been done in the hypertext community in personal history mechanisms. Another factor in the decision was our theory-based belief that memory and the use of memory-related artifacts is a necessarily social process. In a sense, a toolset like Footprints can function as a technological substrate on which community memory operations can be constructed. This has implications for building on-line communities and for efforts to create more functional corporate memory systems.

Adding Identity

Our final reason for keeping Footprints exclusively social was that introducing a notion of identity would imply means of establishing, verifying, and protecting identity and the data associated with it. However, keeping data associated with its creator (the user) also offers potentially significant benefits. As with collaborative filtering, personal history in Footprints suggests that history information could be filtered to be more relevant to the particular user viewing it at any given time. Users might want to view their own histories, perhaps as ways of reviewing their progress over time, recalling lost interests from the past, or retracing steps which were not thought valuable at the time they were taken but that have since gained importance.

Having personal identity within a history-rich system, particularly if the identity was persistent, would enable the representations used by the system to be more personally adapted. One interesting area of adaptation would be landmarks, an element of Lynch's navigation ontology that was not implemented in Footprints. Landmarks are often intensely personal items, reflecting what each of use sees as important in the surrounding environment. A prominent building might be important to someone in a car; a person on foot might find a smaller marker more useful.

A personal history facility would also allow users to create — or the system to determine — elements within the representations that are navigationally important. These elements could be represented more prominently in map and path displays, for example. As with landmarks in the physical world, these prominent elements would help browsers avoid getting lost. In the digital realm they can also perform more active functions than landmarks in the physical world. For example, a user might choose to be notified when a new document that was linked to a landmark appeared in the history database.

Even without personal history, landmarks could be a useful enhancement to the system. In the current implementation, users create landmarks on the fly by clicking on the titles of nodes. These titles stay large and visible as the user moves around the map or path view. The system could be augmented to track which nodes were most

often highlighted in a view and use that information to create landmarks. In essence, the history of use of the history visualization tools would be captured and made available.

Individual identity is also interesting because it can imply authority. If history is, as we argued in Chapter 2, inextricably bound up with expert practice, then the history of what was done by an acknowledged expert could be particularly valuable to someone lacking that expertise. Personalities are also a source of fascination, even when they are not per se experts. We might, for example, be interested in what Web sites Steven Spielberg has visited not because we consider him to be an expert on the Web, but because he is a famous person with eclectic interests and so is accorded more respect.

Adding identity would require that another of Footprints' weak points be addressed: spoofing protection and database manipulation detection. *Spoofing* is the practice of faking one's identity or information for the purposes of manipulating another party. For example, I might spoof my identity to appear as a trusted party in order to get access to data I would not ordinarily be allowed to have. If the system purports to show the history of a notable figure such as Spielberg, then there need to be fairly strong guarantees that the data are indeed Spielberg's. In the case of an amateur spoofing the identity of an expert, the results could be quite serious.

Database *manipulation* is the practice of changing or increasing the data, usually in order to promote or hide a desired target. In today's Web, it is common for sites to liberally "seed" HTML pages with key words or phrases as a form of data manipulation. These seeded words are invisible to users who view the pages — generally the text is held in comments that are not shown in browser software. However, the false keywords cause the page to be more likely to be shown in response to the kinds of keyword-based searches that most Web engines perform. In effect, the data manipulation has made the target document more prominent, perhaps undeservingly.

In the current Footprints system, a malicious person might attempt to repeatedly submit data that point to or include a chosen page in an effort to make that page appear more popular. We make no attempt to block such manipulation in part because there is little incentive for users to do it. A document may show up in a map as "hot" or a trail may show up as "heavily used" but this is true for the most-used quarter of all documents and trails. Once a manipulator has achieved this top quartile, there is no place to go. Footprints does not show the most popular or otherwise promote the top documents. If such a rating system was to be introduced, it would be simple to establish some protection against data manipulation, for example, by implementing something like PHOAKS' one person/one vote system. This assumes, of course, that data manipulation protection is accompanied by protection against identity spoofing so that a malicious person could not repeatedly fake his identity to register more votes.

See "PHOAKS" on
page 99.

Community Identity

History generated by people like the viewer could be more interesting than history generated by the community at large, and might be more easily comprehensible because similar people might share similar beliefs, goals, attitudes, and so on. People with such shared cognitive structures form various kinds of communities. Footprints in its current implementation assumes that communities of interest exist around Web sites — that is, those people who go to a car-buying site form a community of interest by virtue of their taking the time and energy to visit that site. However, another view is that this community exists even before it expresses itself by visiting the site. If this community could be located and connected, it could share knowledge that might be beneficial to its members but irrelevant to people outside the community. This is,

after all, the purpose of professional conferences and trade shows that are not open to the general public. Interaction history could help such communities operate, assuming that the system had the means beforehand to identify a user's possible community memberships.

In summary, identifying users would require additional protections but would offer benefits in that personal and group identities could be used to filter and enhance history data. Identity filtering might also offer a natural way to scale the system: instead of continuing to include all users in one database, separate databases could be established and searched for individual users and communities of users.

The list of possible improvements discussed above is not intended to detract from the accomplishments of this dissertation. Rather, I think they point to the fact that interaction history is a potentially rich and complex area. It is ripe with opportunities for application: Web tools, electronic commerce, and organizational memories are obvious areas for further investigation.

6.2 Conclusion

The tools built as part of this dissertation are experiments; they have uses, but they also suggest ideas for new experiments to follow on to these. Subjects such as dwell time, alternative visualizations of interaction history, and integration with other kinds of systems are now open for investigation. The value of this dissertation lies in the theoretical ideas, not in the specific implemented tools. Where previously work in the field consisted of unconnected clever implementations we now have a framework to use in talking about both the existing tools and about tools we might build for the future. We can see areas where interaction history is an interesting problem-solving technology and we can begin to consider how a digital world might be constructed — a world in which all data would come with a history, as it does in the physical world, might be taken advantage of for solving problems.

APPENDIX A *Subject Instructions*

The following is a copy of the instructions given to subjects participating in the second phase of the experiment. The instructions for the first group of subjects was substantially similar, but omitted mention of the Footprints tools (the second and third paragraphs under the Task section).

Pre-Task Instructions and Waiver

Thank you for agreeing to help us by participating in our data gathering. This form describes an experimental task which we would like you to complete. How you go about doing it is your own choice — there is no right or wrong way to do the task. Before you begin, we will ask you to fill out a form giving us some information about the type of person you are. After you have completed the task we will ask you to fill out another form to get your opinions and feedback on what you have done. If anything you read or are told is unclear, please stop us and ask for clarification.

The Task

The task is to use the Web to investigate a possible automobile purchase. Imagine that you are able to spend approximately \$20,000 on a car; your goal is to figure out what makes/models of automobiles interest you in that price range. You will have 20 minutes to do your browsing. This is not because we want you to rush through things, but because we need to establish some time limit.

During the task you will be able to use some tools from a system called “Footprints” to help you. The tools are *maps* (high-level views), *trails* (lower-level views) and *annotations* (marks on the Web pages themselves).

Before you begin, you will be given a guided practice session which will show you how the Footprints tools may be used. You are **not** required to use these tools during the experiment. They are there for help if you want or need them. You will have a control panel available which will allow you to turn the tools off or on as you desire. You may choose to use some, all, or none of the tools.

Observation

During the browsing, our software will be watching where you go, which selections you make, and so on. These data will be used in later stages of our development. All data will be recorded anonymously — that is, no link will be made between the data and you. Furthermore, all data from all subjects will be combined in such a way that no one will be able to extract which data are yours.

Waiver

By signing below, I acknowledge that I am participating voluntarily, and that I agree to release my data for research and experimental purposes, on the condition that such data are never identified as belonging to me in any direct way. I understand that I may withdraw my participation at any time during the process and that all data related to my participation will immediately be destroyed. I understand that if I feel I have been treated unfairly I may call the MIT Committee on the Use of Humans as Experimental Subjects at 253-6787.

signature

Printed name

date

APPENDIX B *Subject Feedback Form*

The following is a copy of the survey form given to the first group of subjects after they had completed the car-shopping task.

Post-Task Evaluation

Imagine that someone else is going to try to complete the same task as you have just completed. You will not be able to be here to help them, but you might leave some record of what you have done for them to use.

Below are some possible things you might leave. Please rate them on how useful you think they would be, based on your experience of trying this task. There are no right or wrong answers; the purpose of asking these questions is to get your honest opinion.

Item	very useful				not at all useful
What pages you visited	1	2	3	4	5
What pages you did not visit	1	2	3	4	5
Which links did you use	1	2	3	4	5
Which links did you not use	1	2	3	4	5
How many times did you visit a page (relative to other pages)	1	2	3	4	5
How many times did you visit a page (absolute numbers)	1	2	3	4	5
How many times did you traverse a link (relative to other links)	1	2	3	4	5
How many times did you traverse a link (absolute numbers)	1	2	3	4	5
How recently have you visited each node	1	2	3	4	5
How recently have you traversed each link	1	2	3	4	5
How long did you stay at a page	1	2	3	4	5
How long did you stay at a location in a sequence of pages	1	2	3	4	5
What are the "hot" items (based on raw number of visits)	1	2	3	4	5
What are the "hot" items (based on relative number of visits)	1	2	3	4	5
What are the "hot" items (based on recency of use)	1	2	3	4	5

The following items would require the system to ask you for additional information. Again, please rate them on how useful you feel they would be to someone trying to complete the same task as you just completed.

Item	very useful				not at all useful
Who you are (based on your login name or a pseudonym)	1	2	3	4	5
What computer did you connect from	1	2	3	4	5
What kind of site did you connect from (edu, com, etc)	1	2	3	4	5
What type of user you are (casual browser, teacher, corporate researcher)	1	2	3	4	5
Why did you go to a particular page	1	2	3	4	5
What did you think of particular pages you visited	1	2	3	4	5

Finally, of these twenty-one items, which three would you expect to be the most useful:

- 1.
- 2.
- 3.

Thank you for your participation.

APPENDIX C *Post-Test Questionnaire*

The following is a copy of the questionnaire filled out by subjects before and after the task. In the original, it was printed on two sides of a sheet of paper. Subjects filled out the front — describing themselves and their preferences — before completing the task. The back was completed after the task.

Questionnaire

Note that this has two parts. This part, side 1, should be filled in before you do the experiment. Part 2 is on the other side and should only be filled in once you're done since it asks about what you did.

The first few questions are about yourself and your skills/background; please circle your answer:

Your sex: Male Female

Is English your first language: Yes No

Have you shopped for a car in real life? Yes, recently Yes, not recently No

Have you ever looked for car information on the Web before today? Yes No

Have you ever shopped for another purchase on the Web Yes No

If yes, what was your most recent Web purchase:

How expert are you in using the Web: Expert Intermediate Beginner

The next questions are about how you like to get around.

Imagine that you need to go to several different places, none of which you have ever been to before. You can go to them in any order and it's not necessarily easier to get to any of them from any other; it could be random. Given this situation, please circle your answers to the questions below:

If you could only have one, would you prefer: a map written directions

If you could only have one, would you prefer: spoken directions while you were going written directions beforehand

If you could only have one, would you prefer a map annotated by someone who had gone to these places or directions written by someone who had gone to these places

Please do the training session and then the experiment now. When you are done, please answer the questions on the back of this sheet.

The next questions are about what you did in this task:

Please list the car choices (make & model) you found while browsing which met your criteria. If you can not remember all of them, please list as many as you can remember:

How satisfied are you with these choices? (ie, do you think you explored the problem well) Totally satisfied Totally disappointed
 1 2 3 4 5

How satisfied are you with the experience you had? Totally satisfied Totally frustrated
 1 2 3 4 5

How easy was it to find relevant information? Very easy Very hard
 1 2 3 4 5

How easy was it to understand what you saw? Very easy Very hard
 1 2 3 4 5

How well-structured was what you saw? Well structured Incoherent
 1 2 3 4 5

How distracted were you by other things you saw? Not at all Very distracted
 1 2 3 4 5

References

1. Ackerman, Mark S. & Thomas W. Malone (1990) "Answer Garden: A Tool for Growing Organizational Memory," *Proceedings of the Conference on Office Automation Systems*, ACM Press. Page 17
2. Barrett, Rob & Paul Maglio (1997) "Intermediaries: New Places for Producing and Manipulating Web Content," *WWW7: The 7th Annual Conference on the World Wide Web*, available from <http://www7.dstc.edu.au/home.html> Page 104
3. Bartlett, Frederick (1932) *Remembering: a Study in Experimental Social Psychology*, Cambridge University Press. Page 40
4. Bederson, B., J. Hollan, K. Perlin, J. Meyer, D. Bacon, & G. Furnas (1996) "Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics," *Journal of Visual Languages and Computing*, Volume 7. Page 90
5. Bederson, B., J. Hollan, J. Stewart, D. Rogers, D. Vick, L. Ring, E. Grose, C. Forsythe (1998) "A Zooming Web Browser," in *Human Factors in Web Development*, Ratner, Grose & Forsythe (eds.) Lawrence Erlbaum Associates. Page 91
6. Berners-Lee, Tim (1996) URI IEFT Draft Specification. See http://www.unix.digital.com/demos/freetkit/docs/httpd/HTML/Addressing_index.html Page 55
7. Beyer, Hugh & Karen Holtzblatt (1997) *Contextual Design: Defining Customer-Centered Systems*, Morgan Kaufmann. Page 127
8. Brand, Stewart (1994) *How Buildings Learn: What happens after they're built*, Penguin Books. Pages 20, 31
9. Bush, Vannevar (1945) "As We May Think," originally printed in *The Atlantic Monthly* July 1945. Reprinted in *A History of Personal Workstations*, Adele Goldberg (ed.) 1988, ACM Press. Page 105
10. Castaneda, Carlos (1985) *Teachings of Don Juan: A Yaqui Way of Knowledge*, Pocket Books. Page 89
11. Chalmers, Matthew, K. Rodden & D. Brodbeck (1998) "The Order of Things: Activity-Centred Information Access," *Proceedings of the 7th International World Wide Web Conference (WWW7)*. Pages 95, 108
12. CNN (1997) <http://cnn.com/US/9712/25/lost.museum/> Page 21
13. Conklin, Jeff (1987) "Hypertext: An Introduction and Survey," *IEEE Computer*, September, 1987, IEEE Press. Page 46
14. Dahlbäck, Nils, Kristina Höök, & Marie Sjölander (1996) "Spatial Cognition in the Mind and in the World — the case of hypermedia navigation", *Proceedings of the Eighteenth Annual Meeting of the Cognitive Science Society*, University of California, San Diego. Page 51
15. Dieberger, Andreas (1997) "Supporting Social Navigation on the World-Wide Web," *International Journal of Human-Computer Studies*, Vol. 46. Page 91
16. Dourish, Paul & Matthew Chalmers. "Running Out of Space: Models of Information Navigation," presented at HCI'94, Cambridge, UK. Pages 51, 96
17. Downs, Roger & David Stea (1973) *Image and Environment: Cognitive Mapping and Spatial Behavior*, Aldine Press. Page 45
18. Doyle, Arthur C. (1927) *The Complete Sherlock Holmes*, Doubleday & Co. Pages 32, 32

- Page 40 19. Durkheim, Emile (1893/1984) *The Division of Labour in Society*, W. D. Halls (trans.), MacMillan Press.
- Page 40 20. Durkheim, Emile (1895/1982) *The Rules of Sociological Method*, W. D. Halls (trans.), MacMillan Press.
- Pages 43, 24 21. Engeström, Yrjö, Katherine Brown, Ritva Engeström, & Kirsi Koistinen (1990) "Organizational Forgetting: an Activity-Theoretical Perspective," in *Collective Remembering*, Middleton & Edwards (eds.), Sage Publications.
- Page 23 22. Eno, Brian (1999) "The Revenge of the Intuitive." *WIRED Magazine* issue 7.01.
- Page 45 23. Fairchild, Kim, Greg Meredith & Alan Wexelblat (1989) "The Tourist Artificial Reality," *Proceedings of CHI'89 Conference on Human Factors in Computing Systems*, ACM Press.
- Page 78 24. Foner, Leonard. "Entertaining Agents: a sociological case study," *Proceedings of the First International Conference on Autonomous Agents*, ACM Press, 1997.
- Page 69 25. Furnas, George (1986) "Generalized Fisheye Views," *Proceedings of CHI'86 Conference on Human Factors in Computing Systems*, ACM Press.
- Page 73 26. Furnas, George (1997) "Effective View Navigation," *Proceedings of CHI'97 Conference on Human Factors in Computing Systems*, ACM Press.
- Page 102 27. Furuta, Shipman, Marshall, Brenner & Hsieh (1997) *Hypertext Paths and the World-Wide Web: Experiences with Walden's Paths*, Hypertext'97 Proceedings, ACM Press.
- Page 96 28. Goldberg, David, David Nicholas, Brian Oki, & Douglas Terry (1992) "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, Vol 35#12.
- Page 15 29. Grant, Richard (1987) *Rumors of Spring*, Bantam Spectra.
- Page 22 30. Greenbie, Barrie B. (1981) *Spaces: Dimensions of the Human Landscape*, Yale University Press
- Page 46 31. Halasz, Frank G. (1988) "Reflections on Notecards: Seven Issues for the Next Generation of HyperMedia Systems," *Communications of the ACM* 31(7), ACM Press.
- Page 22 32. Hall, Edward T. (1973) *The Silent Language*, Anchor Press.
- Page 22 33. Hall, Edward T. (1990) *The Hidden Dimension*, Anchor Press.
- Pages 19, 24, 89 34. Hill, Hollan, Wroblewski & McCandles (1992) "Edit Wear and Read Wear," *Proceedings of CHI'92 Conference on Human Factors in Computing Systems*, ACM Press.
- Page 19, 89 35. Hill, Will & Jim Hollan (1993) "History-Enriched Digital Objects," *Proceedings of Computers, Freedom, and Privacy (CFP'93)*, available from <http://www.cpsr.org/dox/conferences/cfp93/hill-hollan.html>
- Page 99 36. Hill, Will & Loren Terveen (1994) "New Uses and Abuses of Interaction History: Help Form the Research Agenda" Workshop held at ACM CHI'94, *Proceedings of ACM CHI'94 Conference on Human Factors in Computing Systems*, ACM Press.
- Page 99 37. Hill, Will & Loren Terveen (1996) "Using Frequency-of-Mention in Public Conversations for Social Filtering," *Proceedings of ACM CSCW'96 Conference on Computer-Supported Cooperative Work*, ACM Press.

38. Hutchins, Edwin (1988) "The Technology of team navigation," in *Intellectual Teamwork: Social and Technical Bases of Collaborative Work*, Galegher, Kraut & Egido (eds.), Lawrence Erlbaum Associates. Page 41
39. Hutchins, Edwin (1995) *Cognition in the Wild*, MIT Press. Pages 42, 35, 46, 50, 53
40. Koenemann, Jurgen & Belkin, Nicholas (1996) "A case for interaction: A study of interactive information retrieval behavior and effectiveness," *Proceedings of CHI'96 Conference on Human Factors in Computing Systems*, ACM Press. Page 50
41. Krier, Leonard (1984) "Houses, Palaces, Cities," *A. D. Profile 54*, London, Architectural Design. Page 20
42. Lakoff, George & Mark Johnson (1980) *Metaphors We Live By*, University of Chicago Press. Page 50
43. Lamping, John, Ramana Rao & Peter Pirolli (1995) "A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies," *Proceedings of ACM CHI 95 Conference on Human Factors in Computing Systems*, ACM Press. Page 69
44. Lynch, Kevin (1960) *The Image of the City*, MIT Press. Pages 45, 47, 49, 50
45. Lynch, Kevin (1972) *What Time is this Place?*, MIT Press. Page
46. Machado, Antonio (1875-1939) quoted in *The Harvest of a Quiet Eye: A Selection of Scientific Quotations*, Alan L. Mackay (compiler), Adam Hilger Publications. Page 107
47. Maes, Pattie. "Agents that Reduce Work and Information Overload," *Communications of the ACM*, Vol 37#7, ACM Press. Page 79
48. Maxwell, Joseph A. (1996) *Qualitative Research Design*, Applied Social Research Methods Series, Volume 41, Sage Publications. Page 19
49. Middleton & Edwards, eds. (1990) *Collective Remembering*, Sage Publications. Page
50. Morita, Masahiro & Yoichi Shinoda (1994) "Information Filtering Based on User Behavior Analysis and Best Match Text Retrieval," *Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM Press. Pages 65, 118
51. Neruda, Pablo (1974) "Toward an Impure Poetry", in *Pablo Neruda: five decades*, Grove Press Page 125
52. Norman, Donald A. (1988) *The Psychology of Everyday Things*, Basic Books. Page 27
53. Oren, Salomon, Kreitman & Don (1990) "Guides: Characterizing the Interface," *The Art of Human-Computer Interface Design*, Brenda Laurel (ed.), Addison Wesley. Page 101
54. Orr, Julian E. (1996) *Talking About Machines: An Ethnography of a Modern Job*, IIR Press. Pages 27, 35
55. Orr, Julian E. (1990) "Sharing Knowledge, Celebrating Identity: Community Memory in a Service Culture," in *Collective Remembering*, Middleton & Edwards (eds.), Sage Publications. Page 29
56. Pirolli, Peter (1997) "Computational Models of Information Scent-Following in a Very Large Browsable Text Collection," *Proceedings of ACM CHI 97 Conference on Human Factors in Computing Systems*, ACM Press. Pages 46, 117
57. Pirolli, Peter & Stuart Card (1995) "Information Foraging in Information Access Environments," *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, ACM Press. Pages 26 46

- Page 50 58. Pirolli, Peter, Patricia Schank, Marti Hearst & Christine Diehl (1996) Scatter/Gather Browsing Communicates the Topic Structure of a Very Large Text Collection, *Proceedings of ACM CHI 96 Conference on Human Factors in Computing Systems*, ACM Press.
- Page 26 59. Radley, Alan (1990) "Artefacts, Memory, and a Sense of the Past," in *Collective Remembering*, Middleton & Edwards (eds.), Sage Publications.
- Pages 83, 97 60. Resnick, Iacovou, Suchak, Bergstrom & Riedl (1994) "GroupLens: An open architecture for collaborative filtering of netnews," *Proceedings of the Conference on Computer Supported Cooperative Work*, ACM Press.
- Pages 51, 96 61. Resnick, Paul & Hal Varian, eds. (1997) "Recommender Systems," special section in *Communications of the ACM*, March, 1997, ACM Press.
- Page 49 62. Robertson, Mackinlay & Card (1991) "Cone Trees: Animated 3D Visualizations of Hierarchical Information," *Proceedings of CHI'91 Conference on Human Factors in Computing Systems*, ACM Press.
- Pages 49, 79, 86 63. Salton, Gerard (1971) *The SMART System — Experiments in Automatic Document Processing*, Prentice Hall.
- Page 38 64. Schank, Roger C. (1990) *Tell Me a Story*, Charles Scribner's Sons.
- Page 38 65. Schank, Roger C. & Robert P. Abelson (1995) "Knowledge and Memory: the Real Story," *Advances in Social Cognition*, Volume VIII, Lawrence Erlbaum Associates.
- Pages 33, 34, 35 66. Schon, Donald A. (1983) *The Reflective Practitioner: How Professionals Think in Action*, Basic Books.
- Page 92 67. Schütte, Arjan (1998) *Patina: layering history-of-use on digital objects*, S. M. Thesis, MIT Program in Media Arts & Sciences.
- Page 93 68. Seligman, Doree & Stephan Bugaj (1997) "Live Web Stationery: Virtual Paper Aging," demonstration presented at ACM SIGGRAPH'97.
- Pages 98, 100 69. Shardanand, Upendra (1994) *Social Information Filtering for Music Recommendation*, S.M. Thesis, Program in Media Arts and Sciences.
- Pages 96, 97 70. Shardanand, Upendra & Pattie Maes (1995) "Social Information Filtering: Algorithms for Automating 'Word of Mouth'," *Proceedings of CHI'95 Conference on Human Factors in Computing Systems*, ACM Press.
- Page 34 71. Simon, Herbert (1957) *Models of Man*, John Wiley and Sons.
- Page 34 72. Simon, Herbert (1979) "From Substantive to Procedural Rationality," in *Philosophy and Economic Theory*, F. Hahn and M. Hollis (eds.), Oxford University Press.
- Page 91 73. Smith, Randy B. (1987) "Experiences with the Alternate Reality Kit: An Exaple of the Tension Between Literalism and Magic," *Proceedings of CHI+GI'87 Conference on Human Factors in Computing Systems*, ACM Press.
- Pages 73, 91 74. Spertus, Ellen (1997) "ParaSite: Mining Structural Information on the Web," *Proceedings of the Sixth International World Wide Web Conference*, MIT Press.
- Page 46 75. Stephens, David W. & John R. Krebs (1986) *Foraging Theory*, Princeton University Press.
- Pages 29, 36-37 76. Suchman, Lucy A. (1987) *Plans and Situated Actions: The problem of human-machine communication*, Cambridge University Press.

77. Terveen, Hill, Amento, McDonald & Creter (1997) "Building Task-Specific Interfaces to High Volume Conversational Data," *Proceedings of CHI'97 Conference on Human Factors in Computing Systems*, ACM Press. Pages 17, 99
78. Tufte, Edward (1983) *The Visual Display of Quantitative Information*, Graphics Press. Page 26
79. Wertsch, J. V. (1987) "Collective Memory issues from a socio-historical perspective," *Quarterly Newsletter of the Laboratory of Comparative Human Cognition*, Vol 9#1. Page 41
80. Wexelblat, Alan (1998) "Communities through Time: Using History for Social Navigation," *Community Computing and Support Systems*, Toru Ishida (ed.), Lecture Notes in Computer Science 1519, Springer Verlag. Page 68
81. Wexelblat, Alan (1999) "History-Based Tools for Navigation," *Proceedings of HICSS-99, the 32nd IEEE Hawaii International Conference on System Sciences*, IEEE Press. Page 68
82. Wexelblat, Alan & Pattie Maes (1997) "Visualizing Histories for Web Browsing," *RIAO'97: Computer-Assisted Information Retrieval on the Internet*, Montreal. Page 68
83. Wolff, Michael. (1998) *Burn Rate: how I survived the gold rush years on the Internet*, Simon & Schuster. Page 37
84. Zacharia, Giorgos & Pattie Maes (1999) "Collaborative Reputation Mechanisms in Electronic Commerce," *Proceedings of HICSS-99, the 32nd IEEE Hawaii International Conference on System Sciences*, IEEE Press. Page 52
85. Zellweger, Polle T. (1989) "Scripted Documents: A Hypermedia Path Mechanism," *Hypertext'89 Proceedings*, ACM Press. Page 101

A

alidade 42
American Medical Association 52
Apple Finder 49
Apple Macintosh 75, 101

B

Barrett, Rob 104
Bartlett, Frederick 40
Bederson, Ben 90, 91, 127
Belkin, Nicholas 50
BMW 120
Boston 28, 39
Brand, Stuart 20, 31
Bugaj, Stephan 93
Bush, Vannevar 105

C

Card, Stu 46
carrying capacity 26, 127, 128
Castaneda, Carlos 89
Chalmers, Matthew 51, 95, 108
Conklin, Jeff 46

D

Dahlbäck, Nils 51
Dieberger, Andreas 91, 92
Dourish, Paul 51
Downs, Roger 45, 46, 47
Doyle, Sir Arthur Conan 32, 33
Durkheim, Emile 40

E

Edmund's 120
Emacs 21, 22, 25
Emerson, Ralph Waldo 45
Engeström, Yrjö 24, 43
Eno, Brian 23

F

Furnas, George 73
Furuta, Richard 102

G

Goldberg, David 96
Grant, Richard 15
Greenbie, Barrie 22

H

Halasz, Frank 46
Hall, Edward 22
Hill, Will 19, 24, 89, 90, 111
Hollan, Jim 19, 24, 89, 90
Holmes, Sherlock 32, 33, 38
Hutchins, Edwin 35, 41, 42, 45, 50, 51, 53

I

IBM 103, 104
information retrieval 15
information scent 26, 117, 118, 127
Infoseek 120
intelligence, collective 17

J

Java 15, 53, 56, 58, 61, 62, 80, 128
Javascript 53, 58, 80
JIOMA 62
Johnson, Mark 50

K

Kennedy, John F. 41
Klock, Felix 61
Koenemann, Jurgen 50
Krebs, John 46
Krier, Leon 20

L

Lakoff, George 50
Lamping, John 69
Lian, Alex 65
Lynch, Kevin 45, 47, 48, 49, 50, 53, 71, 74, 129

M

Machado, Antonio 107
Maes, Pattie 52, 79, 80
Maglio, Paul 104
Maxwell, Joseph 19
Michigan 17
Micronesia 51
Microsoft Explorer 53
Microsoft Office 29
Microsoft Windows 49, 75
MIT 116
MIT Media Lab 15, 48, 66, 100, 128
Morita, Masahiro 65

N

navigation 17, 18, 24, 42, 89, 90, 91, 100, 101, 102, 118, 128, 129
navigation, social 18, 39, 51, 89, 95, 96, 105, 126
Neruda, Pablo 125
Netscape 94, 95
Netscape Navigator 22, 53, 112
Nicholas, David 96
Nixon, Richard 41
Norman, Donald 27

O

Oren, Tim 101
Orr, Julian 27, 35
Orwell, George 42

P

Park Street Station 39
Perl 51
Pirolli, Peter 25, 46, 50, 117

R

Radley, Alan 26, 40
Revere, Paul 28
Robertson, George 49
Ruth, Babe 28

S

Salton, Gerard 49
satisfice 28, 34, 35, 98
Schank, Roger 36, 38, 39, 44
Schön, Donald 33, 34, 35, 36, 37, 44

Schütte, Arjan 92, 93
Seligman, Doree 93
Shinoda, Yoichi 65
Simon, Herb 34
Simpson, O. J. 16
Sistine Chapel 31
Smith, Randy 91
Space Shuttle 41
Sphinx 31
Spielberg, Steven 130
Starseed 102
Stea, David 45, 46, 47
Stephens, David 46
Suchman, Lucy 29, 36, 37, 38, 44

T

Texas A&M University 102
Tufte, Edward 26

W

Washington, D.C. 41
Washington, George 28
Wertsch, J. V. 41

X

Xerox PARC 26, 46, 49, 50, 69, 96

Y

Yahoo 51, 52, 120

Z

Zellweger, Polle 101