

# Control Choices and Network Effects in Hypertext Systems

*E. James Whitehead, Jr.*

Dept. of Information and Computer Science  
University of California, Irvine  
Irvine, CA 92697-3425  
Tel: +1 949 824 4121  
E-mail: ejw@ics.uci.edu

## ABSTRACT

When the utility of a hypertext system depends on the number of users and amount of data in the system, the system exhibits *network effects*. This paper examines how the core differences in control assumptions between monolithic hypertext systems, open hypermedia systems, and the Web, lead to different incentive structures for readers and content providers and hence varying levels of network effects.

Significant results of this analysis are as follows. First, lack of control over the data in a hypermedia system, combined with a large-scale distribution infrastructure is a key aspect of achieving network effects, since this control choice affords large numbers of readers. Second, examination of network effects from the Web and monolithic hypermedia systems suggests that control over the user interface is a key contributor to network effects, since it provides a more pleasant experience for readers, and allows for more control over the presentation by content providers. Finally, control over the hypermedia structure provides a negative contribution to network effects, since the control point limits scalability, thus capping the total number of readers.

**KEYWORDS:** network effects, architectural control choices, monolithic hypertext, open hypertext, WWW

## INTRODUCTION

Hypertext systems exhibit *network effects* [24], where the utility of the hypertext system depends on the number of users and amount of data in the system. This differs markedly from traditional goods whose utility does not depend on the number of other people who also own the good. Breakfast cereal has the same usefulness whether one or a million people own it. In contrast, one telephone has no utility, and two have not much more. But when a million, or a billion people have telephones, the utility of a single telephone is immense.

The World Wide Web has been widely adopted, and is in daily use worldwide by millions of people. More information and services are available on the Web now than ever before, orders of magnitude more than just a few years ago. For an individual user of the Web, the Web is more useful and more valuable today, with millions of other people using it, than it was in 1993-4 when the user base was much, much smaller. Put another way, the utility of a Web browser increases as other people adopt Web browsers, and begin using the Web, without any further investment in the existing browser. Clearly there are network effects present in the Web.

While the network effects present in the Web are related to the number of people using the Web, their importance in the adoption of the Web raise the issue of exactly how the Web generates network effects. Since the Web is a hypertext system, whether other hypertext systems also generate network effects the same way is of interest. The first section of this paper addresses these issues, providing a model for how readers and content providers generate network effects via an incentive feedback loop through the corpus of information and services provided by a hypertext system. This model is shown to be applicable to all hypertext systems.

Since existing research and commercial hypertext systems can also generate network effects, this raises the issue of why these systems haven't been as widely adopted as the Web. For the earliest hypertext systems, later termed "monolithic", the goal was to create a system that provided hypertext functionality. Open hypermedia systems and the Web went further, wanting to provide hypermedia functionality in more open contexts than could be addressed by the monolithic systems. Due to these differing goals, each class of hypermedia system made different control decisions in its architecture. Monolithic hypertext systems control the user interface, hypermedia structure, and data, while open hypermedia systems control the hypermedia structure, and sometimes the data, but relinquish control over the user interface. With the Web, the user interface is controlled via the browser, but the data and hypermedia structure are uncontrolled.

This paper examines how the core differences in control

assumptions between monolithic hypertext systems, open hypermedia systems, and the Web, lead to different incentive structures for readers and content providers and hence varying levels of network effects. In particular, control over system data and its implications for distribution and wide-area access to this data are a key determinant of a hypertext system's ability to generate network effects.

### **A MODEL OF THE GENERATION OF NETWORK EFFECTS**

Observers of the Web intuitively know the utility of the Web has grown with its user population. However, is the utility of the Web directly related to the number of people using it, or is it more the case that this utility is related to an increase in the amount of information available on the Web? The question boils down to describing exactly how the Web generates its network effects. This section gives a model for how the Web creates network effects based on interactions between readers and content providers. Furthermore, this model is shown to have general applicability to all hypertext systems by noting parallels between the Web and existing hypertext systems, and by showing that hypertext systems are instances of hardware/software systems. This model is employed in the rest of the paper to examine how existing hypertext systems generate and often limit network effects.

To examine how a hypertext system generates network effects, it is useful to examine two sets of users, the readers, and the content providers. A reader uses the hypertext system in a read-only fashion, viewing information and using services provided by the system. In contrast, a content provider uses the hypertext system in a read/write manner, both providing new content while also reading information and using services on the system. So, while all users of a hypertext system are considered to be viewing content, a subset of the users also adds (and removes) information to the system. Since it is possible that all of the users are content providers, this division of users still applies to systems such as NoteCards [26], KMS [1], and Intermedia [28] which blur the distinction between readers and content providers.

From the perspective of a reader, the utility of a hypertext system is directly related to the amount of *information and services* available on that system. This is different from the telephone system, where the utility of each telephone is directly related to the number of other telephone users, and not to the amount of data flowing through the telephone network. Readers also derive utility from hypertext links, since a link between two documents increases the usefulness of *both* documents by making a relationship between the documents explicit, and reducing the burden of retrieving the associated information. As the number of links increases, the utility of documents in a hypertext system is greater than those outside the system. In globally distributed hypertext systems, the freshness of information in the system increases the utility of documents over ones outside the system. Services such as shopping and searching also increase hypertext system utility for readers.

Data from the characterization of Web users [13] provides

support for the assertion that reader utility is related to the amount of information and services. The excellent Web survey conducted by Georgia Tech from January, 1994, through 1998, provides a wealth of information on Web usage trends. One finding from these surveys is that information seeking is a primary activity among Web users. Across all surveys, Web users accessed the system frequently, with over 80% of survey respondents using the Web once or more a day (growing over time from 80% in 1994 to 88% in 1998) indicating that users find the Web, and presumably the information on the Web, to be very useful. The surveys differentiate information seeking goals, with the 1998 survey showing significant use of the Web to gather information for education, entertainment, work, and personal information. A low but growing percentage of users are shopping on the Web, employing shopping services.

Other hypertext systems demonstrate the same trends, with many HyperCard stacks available on the Web having educational content. A listing of Hyperties Webs in [25] shows webs containing educational and research content. The hypertext webs purveyed by Eastgate Systems [10] on the HyperCard and StorySpace platforms are primarily for educational and entertainment uses.

From the perspective of content providers, the utility of each document or service in the system is related to the number of people who read or use it. Assuming uniform popularity of information, as more people use the system, more people will access a given piece of information or service. As the utility of information increases within the hypertext system, there is incentive to provide more information and services.

Unfortunately there is a paucity of published data concerning the motivations of content providers. An ad-hoc sampling of Web pages returned by an AltaVista query of "why publish on the Web" yields these commonly occurring motives:

- Access to a global audience,
- Information is instantly available without using a physical publishing process,
- Links to the information can be created by the content provider, and by others,
- Information is more valuable in electronic form, since it is easy to print, quote, email, repurpose, etc.,
- Content providers can receive (potentially rapid) feedback on their information.

There was also a common observation that the Web has a low cost to publish, reflecting that for many content providers, the cost of publishing on the Web is either a sunk cost, due to prior purchase of server computers and an existing network connection, or an external cost, due to network access appearing in the cost structure of another organization, such as a computer support group. This low cost combined with high utility provides significant incentive for adoption of the Web as a medium for publishing content.

Simple feedback loops lead to increases in readers and content providers. Readers are lured to the system by the content and services provided by the hypertext system, as well as by the greater utility of information within the system. As more readers use the system, content providers have incentive to add more content. Content entices readers, readers attract content, and so on. Thus, so long as the amount of information being added is greater than the amount of information being removed, the increasing corpus of information will lead to an increase in the number of readers and content providers. This assertion rests on many assumptions: that the cost of becoming a reader (e.g., getting a Web browser) is uniform over time, that the cost of adding information to the system is uniform over time, and that no external factors limit the total number of readers or content providers.

Another complication for this model is the observed fact that web content is not uniformly popular. Pitkow notes in [21] that two separate studies show roughly 25% of the Web servers account for 85% of the traffic observed at any one Web proxy. But, notably, the most popular sites observed at one proxy are not the same ones observed at another. Additionally, [21] also notes that the average life span of a document on the Web is approximately 50 days, with HTML resources being modified more frequently than images or other media. This strongly suggests that freshness of information is a major incentive for readers to use a hypertext system, and may deserve a more prominent place in the model for network effects creation.

For hypertext systems where network effects are present, once they pass a critical threshold, the feedback cycle of readers and content providers causes users to be *locked-in* to the hypertext system as competing systems are unable to generate sufficient network effects to supplant the dominant system. Such is the case with the World Wide Web today.

#### **Hypertext Systems as Hardware/Software Systems**

Within the economic literature, *hardware/software* systems [15] are those where the consumer must purchase some durable good (the hardware) in order to gain access to information available in a format compatible with the device (the software). An example of a hardware/software system is a video cassette recorder (VCR), where the VCR constitutes the hardware and films on VCR tapes are the software. When economies of scale are present in the production of such compatible software (e.g., the sale and rental of VCR tapes), the amount of available software will be directly related to the number of hardware units sold (e.g., the total number of VCRs in use). That is, as more hardware is sold, there is greater incentive to produce compatible software. Due to this, consumers will base their hardware purchase decision on their perception of the present *and future* availability of compatible software. Over time as more software becomes available, the utility of each hardware device increases, including ones previously sold. Intriguingly, when economies of scale are present in the production of both hardware and software for a given system, the per-unit cost of hardware and software decreases as the overall value of the system increases. In these cases, a significant portion of the value of these systems is not

captured in the price of either the hardware or the software, and hence these network effects are termed *network externalities* [16]. Users of such systems often capture this uninternalized value.

Since readers of a hypertext system derive their utility from the information and services provided by the system, hypertext systems are similar to hardware/software systems in their generation of network effects. For hypertext systems, the “hardware” portion of the system is the program (or programs) used to access hypertext-linked information, which in this analogy is the “software”. So, for the Web, a Web browser is the “hardware”, and Web pages, images, scripts, and Java programs are the “software”. This analogy extends to other hypertext systems as well, for example, in HyperCard, the reader is the “hardware” while HyperCard stacks are the “software”. Like other hardware/software systems, hypertext systems generate network effects due to the interplay between numbers of “hardware” and amount of available “software”; as the amount of information increases, the utility of each instance of hypertext browsing software increases. With economies of scale, this then leads to increases in the amount of information available for viewing on each hypertext browser.

#### **MONOLITHIC HYPERTEXT SYSTEMS, CONTROL CHOICES AND NETWORK EFFECTS**

Monolithic hypertext systems, such as Intermedia [28], KMS [1], NoteCards, [26], HyperCard [4], and StorySpace [7][9] (a non-exhaustive, but representative list), were motivated by a desire to keep their information base internally consistent, and to provide a consistent user interface. As a result, these systems have an architecture which tightly controls the data, hypertext structure, and user interface of the system. Hypertext readers using these systems have a nice user experience with fast link traversals, and few, if any, broken links. Content providers using monolithic hypertext systems are required to import data into the system, or use system specific editors. Data storage is typically limited to a single file or database (e.g. HyperCard, NoteCards) stored locally or on a network file system, or to collections of files stored across a network file system (e.g., KMS), limiting the amount of data which can be accommodated by the system, and preventing distribution of the data across a wide area network.

The choice to control all aspects of the system leads to limited network effects. While readers are attracted to these systems by the rich, highly useful content initially provided by the system, the amount and variety of this content is limited. Because there are relatively few initial readers, and due to the need to learn new editors, content providers do not have a lot of incentive to create new content. Since there are no provisions for remote access to the hypermedia content, the population of readers is limited to those who have access to the local file system, or to those who acquire or purchase complete hypertexts packaged on a disk or CD-ROM from the limited existing distribution channels. Thus, though there was sufficient initial interest from readers of these hypertexts, there was insufficient motivation for content providers to add new information, eventually

leading to a lack of interest from readers. No network effects were generated.

The need to provide incentive for content providers was noted in [12], which states:

The use of hypertext and hypermedia systems is still largely confined to the research community. This is partly because of the limitations of commercially available systems and partly because of the tremendous effort required to create and maintain a hypertext system. These issues are compounded by the fact that currently available hypertext packages are basically closed systems, so that if material is created in one system it is very difficult to integrate it with material created in another system. We believe that this is a major barrier to the growth and development of hypertext and hypermedia applications outside the research community. (p. 299)

Hypercard and StorySpace provide some limited exceptions to the lack of network effects for monolithic hypertext systems. Since HyperCard was freely distributed with Macintosh computers for several years, and HyperCard players are still part of the MacOS, a sufficient base of potential readers existed to provide incentive for development of commercial HyperCard stacks. The ability to neatly package a hypertext into an easily transportable unit, the stack, also facilitated the development of commercial HyperCard stacks. By adding commercial incentive to produce content, more content was developed for HyperCard than if all content utility depended solely on the number of people reading free content. But, at the same time, this reduced the likelihood a reader would view the content due to its cost, reducing audience size.

Additionally, for content providers HyperCard does not provide many of the key benefits of the Web. HyperCard information is not instantly available, and in the case of commercial sales, does have to go through a physical publishing process. HyperCard stacks cannot be linked-to from other stacks, while Web pages can link to each other. Furthermore, rapid feedback on a quickly evolving hyperdocument is more difficult with HyperCard than on the Web. As a result, in 1989 Meyrowitz noted that HyperCard is used, "typically not for daily knowledge work, but as a special-purpose [tool] to create online-help or specialized corpuses for a particular problem domain." [18] (p. 107). Today the majority of HyperCard content is educational, produced by educators whose job is to produce content for a small collection of readers (their students), and hence do not require the incentive of a large set of readers or monetary compensation to derive utility from the content.

StorySpace [7][9] is another interesting exception. With StorySpace, content providers use the system as a form of self-expression, and are not motivated primarily by a large readership for their hypertexts, although a commercial market for literary hypertext does exist. The desire for self-expression provides sufficient motivation to produce content, and the value provided by large numbers of readers is not necessary to "prime the pump" to provide incentive

for generation of content. Also, given the artistic requirements for total visual control over the hypertext, a monolithic hypertext system best meets the needs of literary hypertext.

#### **OPEN HYPERMEDIA SYSTEMS, CONTROL CHOICES AND NETWORK EFFECTS**

Open hypermedia systems began as an explicit reaction against the closed nature of monolithic hypertext systems. [20], which describes the first open hypermedia system, begins with a paragraph describing the closed nature of monolithic hypertext systems, and follows with a paragraph extolling the openness of Sun's Link Service. Meyrowitz agrees, stating that the key factor in the lack of adoption of hypertext systems is their insular, monolithic nature which "demand the user disown his or her present computing environment to use the functions of hypertext and hypermedia." [18] (p. 107)

For most open hypermedia systems (a non-exhaustive list includes DHM [14], Hyperform [27], Microcosm [8], Multicard [23], and Chimera [2]), a key quality of openness is the support of the heterogeneous tools which populate a user's computing environment. The rationale for this requirement is generally pragmatic: content is produced by these tools, and they are the locus of work on the computer. To provide hypertext support to the user's environment, hypertext must be brought to the tools, rather than bringing the output of the tools to the hypertext system.

By focusing on adding hypertext functionality to desktop applications, open hypermedia systems consciously relinquish control over the user interface for data in the hypermedia system, and accept the need for an application launcher component to invoke applications as needed after a link traversal. Other control choices vary (for an in-depth description of the various control tradeoffs in open hypermedia systems, see [19]). Link server systems maintain control over the hypertext structure, but also relinquish control over the data being linked, allowing it to reside in multiple repositories. Open hyperbase systems control both the hypertext structure and the data being linked, thus providing greater consistency, but requiring applications to use its data repository.

Unlike monolithic hypertext systems, some designers of open hypermedia systems directly considered network effects. The conclusion of [20] notes:

With an open protocol, the power of each element of a system expands as it interoperates with others. Open linking can make the power of hypertext available to the world of software. We hope to see linking, and attendant hypertext capabilities, as much a standard part of the computer desktop as the cutting and pasting of text are today. (p. 145)

A call to arms is presented in the introduction to [8]:

The next generation of hypermedia must appear to the user as a facility of the operating system that is permanently available to add information linking and

navigation facilities with the minimum amount of user intervention and without subtracting any of the functionality that was previously available. (p. 182)

The analysis of network effects for open hypermedia systems can still be viewed in terms of readers and content providers, but is shifted towards considerations of tool integration because the user interface to the data in the hypermedia system is via pre-existing tools which are generally hypertext unaware. Since many open hypermedia systems have little or no separation between reading and authoring, readers and content providers are often the same.

Readers are motivated to use an open hypermedia system because the system contains documents and other information that they directly interact with, for example a software engineer interacting with the source code and other artifacts of software project. Since readers of open hypermedia systems are usually also authors, they will be drawn to the system, the locus of their work. Additionally, readers benefit from the hypertext linking between related documents. As noted above in [20], hypertext linking increases the utility of each application, due to the interoperation provided by hypertext link traversals. Content providers have incentive to add links because they are immediately useful (i.e., they are in data used by the content producer), or can be traversed by other users of the system. The ability to link together data is limited only by the number of hypertext-aware applications. This realization motivates the desire to provide open hypermedia services in the operating system, since pervasive availability of hypermedia services would lead to more hypertext-aware applications.

Open hypermedia systems have many problems that stem directly from not controlling the user interface and not controlling the hyperlinked data, and these problems limit the ability to generate network effects. The editing problem, the data versioning problem, and difficulties with user interface consistency are noted in [8]. Add to these the difficulty of configuration management of different versions and types of applications across user environments, and the problem of limited screen real estate after several applications have been launched. Finally, the lack of highly scalable remote data access support in open hypermedia systems is also a noted problem which has spawned much current research. Altogether, these issues reduce the incentives for readers, and increase the maintenance burden for content providers. The lack of distribution support further caps the total possible number of readers, putting an upper limit on the potential utility of the information. However, even if global distribution was available, the problems inherent in providing hypertext services across widely divergent user machine and application configurations would also limit the utility of these hypertexts for readers.

#### **WORLD WIDE WEB, CONTROL CHOICES AND NETWORK EFFECTS**

The Web began as a reaction against an information space very similar to that being touted as ideal by the open hypermedia community. Rather than having to use multiple

programs on several computers to access information, the Web aspired to be a unified access point for information provided by these programs. Slides from a 1993 presentation [5] describe the concept of universal readership:

Before W3, typically to find some information at CERN one had to have one of a number of different terminals connected to a number of different computers, and one had to learn a number of different programs to access that data. The W3 principle of universal readership is that once information is available, it should be accessible from any type of computer, in any country, and an (authorized) person should only have to use one simple program to access it.

To achieve this goal, the Web made different control tradeoffs from either monolithic or open hypermedia systems, controlling the user interface (via the browser) but not controlling either the hypertext structure, or the hypertext data. The lack of control over the hypertext structure and data allowed these aspects of the system to be massively decentralized. The triad of standards, URL [6], HTTP [11], and HTML [22] provided the foundation for interoperation in a widely distributed, large-scale information space.

With the clarity of hindsight, the Web appears optimally suited for generating network effects. As the 1993 talk notes:

To allow the web to scale, it was designed without any centralized facility. Anyone can publish information, and anyone (authorized) can read it. There is no central control. To publish data you run a server, and to read data you run a client. All the clients and all the servers are connected to each other by the Internet. The W3 protocols and other standard protocols allow all clients to communicate with all servers.

Since the Web provided a single user interface to existing repositories of information (a valuable interface on the early Web was to the phone book at CERN), as well as hypertext linking from documents which supported HTML, readers had incentive to use the system. For content providers, the Web offers a significant barrier to entry, requiring the installation and configuration of a Web server and, for many providers, initial or improved connection to the Internet. Not surprisingly, the early Web was limited by the small amount of information available, and the fact this information was related almost entirely to high energy particle physics. Two events in 1993 reduced the barriers to entry for both readers and content providers. First, the NCSA HTTP server was released, and was rapidly ported to most current computing platforms. Unlike the other existing server, the CERN server, this server could be installed by any user, and did not require super user (root) access, and this allowed installation of Web servers without the need for securing buy-in from typically conservative computing support organizations. Second, the release of the Mosaic browser on Unix, Mac and PC platforms increased the base of potential users, and provided a visually pleasing interface which increased reader's incentives for using the system. While these two

events would eventually have touched off the frenzy of growth which categorized the Web in 1994-6, an article in the Business section of the New York Times in December, 1993 [17] added sufficient new users to jump-start the cycle of increasing network effects, as new readers increased the incentives for content providers, who provided more information, leading to more readers, etc.

By controlling the user interface, the Web is able to provide a single, attractive, easy-to-use entry point into the system. Recognizing that a single application cannot provide viewers for all media types, the typical browser provides launch-only hypertext services to invoke an application which displays the unknown media type, and plug-ins, which allow viewers for unknown types to use the same screen real estate as the browser. If the Web controlled the hypermedia structure, it would have led to a single scalability choke point as increasing numbers of systems accessed the same system for link information. By not placing control requirements on the data displayed by the system, the Web could accommodate a wide range of information repositories, enabling more information providers.

Though the Web has well-known drawbacks, with broken links, slow data access, and lack of versioning support being more frequently mentioned problems, it is notable that these problems have not created sufficient disincentive for readers or content providers to cause them to abandon the system, nor have they noticeably dampened the rate of adoption of the Web.

#### **Gopher, the Web, and User Interface Control**

A brief comparison with the Gopher distributed information system [3] highlights both the importance and the subtleties of user interface control, specifically whether the system or the content provider controls the presentation of information to readers. The Gopher system was in existence in the early 1990's at the same time as the initial development of the Web. Gopher made the same broad control assumptions as the Web, controlling the user interface, but not controlling the data or the graph structure of the data. Like the Web, Gopher clients employ an Internet protocol (the Gopher protocol) to retrieve information from a remote server, which they then display. Gopher information is organized into a potentially cyclic graph structure where the graph is distributed, with each Gopher server only containing a small portion of the graph. This opens the possibility that the graph may be inconsistent, with some nodes or subtrees potentially unavailable (dangling links). Like the Web, this lack of control allows the Gopher system to be distributed and scalable. In terms of network effects generation, this placed no architectural cap on the total possible number of readers or amount of information.

Gopher's graph structuring of information is directly reflected in a Gopher client's user interface, which presents a list of menu items to the user. Selecting a menu item either retrieves a single document or image, another menu, or accesses a search service. Due to the design of the protocol, menu items can only be text strings, and those only in ASCII since there is no support for other character sets or

language tagging. The menu oriented Gopher user interface is very simple, at once being its greatest virtue, and most significant drawback. The simplicity of Gopher means that readers find it easy to learn, and easy to access information on the system. However, by requiring the use of menus to access information, even when it does not make sense, the type of information that can be placed on the system is limited—a direct consequence of strict user interface control. Information like online magazines, newspapers, and product catalogs just do not map well to a menu structure. The lack of hypertext capability in the leaves of the graph, the actual documents, further limits both the kind and hypertext structure of Gopher information. So, while the simplicity and scale of Gopher initially attracted some readers and content providers, mostly in universities, the strict control over the user interface, and the lack of a presentation language provided a disincentive for content providers.

Once Gopher and the Web came into direct contact, the richer content of the Web was far more capable of generating network effects than the more strictly controlled, yet more simple Gopher user interface. The presentation control afforded by HTML, including the important ability to have compound documents with images, and the structure control afforded by hypertext links, was far more attractive to both readers and content providers, and was more capable of presenting a greater range of content. So, while both Gopher and the Web made identical data control choices in order to have Internet scale, they differ in their form of user interface control. While both Gopher and the Web control their user interface more than an open hypermedia system, Gopher's user interface is strictly controlled, while the Web's is less so, granting the content provider far greater control over the presentation of information. This was a key factor in the Web's ability to generate network effects more rapidly than Gopher, and supplant it for the dissemination of menu-organized information on the Internet.

#### **CONCLUSION**

This paper has described a model of how a feedback loop of readers, content, and content providers leads to the generation of network effects in hypertext systems. Readers are drawn to a hypertext system by the information and services available on the system, and content providers have incentive to provide content as more readers increase the value of content in the system. Three classes of hypertext systems, monolithic, open, and Web were analyzed from the perspective of the control decisions embedded in their architectures, and how these control decisions led to differing levels of network effects.

The discussion in this paper makes several points. First, lack of control over the data in a hypermedia system, combined with a large-scale distribution infrastructure, is a key aspect of achieving network effects, since this control choice affords large numbers of readers. Examination of network effects from the Web and monolithic hypermedia systems suggests that control over the user interface is a key contributor to network effects, since it provides a more pleasant experience for readers, and allows for more control over the presentation by content providers. However, the

Gopher experience shows the importance of giving fine-grain presentation control to content providers while still retaining coarse control over the user interface. Control over the hypermedia structure provides a negative contribution to network effects, since the control point limits scalability, thus capping the total number of readers.

This paper has analyzed the three major classes of hypertext systems from the viewpoint that the generation of network effects is always a positive outcome due to its benefits for system adoption. But, bigger is not always better. Open hypermedia systems still provide superior support for activities like software development and engineering design work where data is local, consistency of the hypertext structure is necessary, and hypertext support for frequently used tools is important. Similarly, for *personal* knowledge work, Internet access to someone's rough thoughts does not immediately seem an advantage. Open hypermedia systems, and hypertext systems for personal knowledge work still have significant utility despite the fact that, at present, the control choices in these systems make them poorly suited for the creation of network effects.

#### ACKNOWLEDGMENTS

Phil Agre introduced me to the economic literature on network effects, and he, along with Rohit Khare provided feedback on ideas for this paper. Comments and encouragement on an earlier version of this paper from participants at the 4<sup>th</sup> Workshop on Open Hypermedia Systems were very valuable. Comments from the reviewers were especially helpful in strengthening this paper.

#### REFERENCES

1. R. Akscyn, D. L. McCracken, E. Yoder, KMS: A distributed hypermedia system for sharing knowledge in organizations, *Comm. ACM*, 31(7), 820-835, July, 1988.
2. K. M. Anderson, R. N. Taylor, E. J. Whitehead, Jr., Chimera: Hypertext for heterogeneous software environments, *Proc. ECHT'94*, Edinburgh, Scotland, September, 1994, pages 94-107.
3. F. Anklesaria, M. McCahill, P. Lindner, D. Johnson, D. Torrey, B. Alberti, "The Internet Gopher Protocol (a distributed document search and retrieval protocol)," University of Minnesota, RFC 1436, March, 1993.
4. Apple Computer, Inc., *Hypercard User's Guide*, Cupertino, California, 1987.
5. T. Berners-Lee, World Wide Web Seminar, unpublished slides, <http://www.w3.org/Talks/General.html>.
6. T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax," MIT/LCS, U.C. Irvine, Xerox, Internet Draft Standard RFC 2396, August, 1998.
7. J. D. Bolter, M. Joyce, Hypertext and Creative Writing, *Proc. Hypertext'87*, Baltimore, 1987, pages 41-50.
8. H. Davis, W. Hall, I. Heath, G. Hill, Towards an integrated information environment with open hypermedia systems, *Proc. ECHT'92*, Milano, Italy, November-December, 1992, pages 181-190.
9. Eastgate Systems, Inc., *Storyspace* hypertext writing environment for Macintosh computers, 1991.
10. Eastgate Systems, Inc., <http://www.eastgate.com/>
11. R. Fielding, J. Gettys, J. Mogul, H. Frystyk, T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1," U.C. Irvine, DEC, MIT/LCS, Internet RFC 2068, January, 1997.
12. A. M. Fountain, W. Hall, I. Heath, H. Davis, Microcosm, An open model for hypermedia with dynamic linking, *Proc. ECHT'90*, Versailles, France, November, 1990, pages 298-311.
13. GVU Web Survey Team, GVU's WWW User Surveys, Georgia Institute of Technology, [http://www.cc.gatech.edu/gvu/user\\_surveys/](http://www.cc.gatech.edu/gvu/user_surveys/)
14. K. Gronbæk, J. A. Hem, O. L. Madsen, L. Sloth, Designing Dexter-based cooperative hypermedia systems, *Proc. Hypertext'93*, Seattle, Washington, November, 1993, pages 25-38.
15. M. L. Katz, C. Shapiro, Systems competition and network effects, *Journal of Economic Perspectives*, vol. 8, no. 2, 1994, pages 93-115.
16. M. L. Katz, C. Shapiro, Technology adoption in the presence of network externalities, *Journal of Political Economy*, vol. 94, no. 4, 1986, pages 822-841.
17. J. Markoff, A free and simple computer link; enormous stores of data are just a click away, *New York Times*, v143, Wed, Dec 8, 1993, col 3.
18. N. Meyrowitz, The missing link: why we're all doing hypertext wrong, in E. Barrett, ed., *The Society of Text*, MIT Press, Cambridge, Mass., 1989, pages 107-114.
19. K. Østerbye, U. K. Wiil, The Flag taxonomy of open hypermedia systems, *Proc. Hypertext'96*, Washington, DC, March, 1996, pages 129-139.
20. A. Pearl, Sun's Link Service: A protocol for open linking, *Proc. Hypertext'89*, Pittsburgh, Pennsylvania, November, 1989, pages 137-146.
21. J. E. Pitkow, Summary of WWW characterizations, In *Proc. 7<sup>th</sup> Int'l WWW Conference*, Brisbane, Australia, April 14-18, 1998, published as *Computer Networks and ISDN Systems*, vol. 30, nos. 1-7, April, 1998, pages 551-558.
22. D. Raggett, A. Le Hors, I. Jacobs, "HTML 4.0 Specification," W3C Recommendation REC-html40-19980424, April, 1998.
23. A. Rizk, L. Sauter, Multicard: An open hypermedia system, *Proc. ECHT'92*, Milano, Italy, November-

December, 1992, pages 4-10.

24. J. Rohlfs, A theory of interdependent demand for a communications service, *Bell Journal of Economics* 5(1), 1974, pages 16-37.
25. B. Shneiderman, Reflections on authoring, editing, and managing hypertext, in E. Barrett, ed., *The Society of Text*, MIT Press, Cambridge, Mass., 1989, pages 115-131.
26. R. H. Trigg, L. Suchman, F. G. Halasz, Supporting Collaboration in NoteCards, *Proc. Computer-Supported Cooperative Work (CSCW'86)*, Austin, Texas, pages 153-162.
27. U. K. Wiil, J. J. Leggett, Hyperform: Using extensibility to develop dynamic, open and distributed hypertext systems, *Proc. ECHT'92*, Milano, Italy, November-December, 1992, pages 251-261.
28. N. Yankelovich, B. Haan, N. Meyrowitz, S. Drucker, Intermedia: the concept and the construction of a seamless information environment. *IEEE Computer*, 21(1):81-96, January, 1988.