

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC and YOUTUBE, LLC
Petitioners

v.

ROBOCAST, INC.
Patent Owner

Case IPR2023-00592 & IPR2023-00593
U.S. Patent No. 8,965,932 B2
Issue Date: February 24, 2015

Title: AUTOMATED CONTENT SCHEDULER AND DISPLAYER

DECLARATION OF CHRISTOPHER M. SCHMANDT

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

TABLE OF CONTENTS

	Page
I. INTRODUCTION AND QUALIFICATIONS	8
A. Qualifications and Experience	8
B. Materials Considered.....	12
II. PERSON OF ORDINARY SKILL IN THE ART	14
III. STATEMENT OF LEGAL PRINCIPLES.....	17
A. Claim Construction	17
B. Obviousness.....	19
IV. THE '932 PATENT.....	25
A. Overview of the Specification.....	25
B. The Challenged Claims	28
V. APPLICATION OF THE PRIOR ART TO CHALLENGED CLAIMS	29
A. Brief Summary of Prior Art	30
1. References Common to All Grounds.....	30
(a) Shimizu (EX1003).....	30
(b) Benedict (EX1006).....	39
(c) Fox (EX1005).....	43
(d) Wolfe (EX1014)	45
2. Additional References.....	49
(a) Guinan (EX1008).....	49
(b) Miller (EX1010)	52
(c) Filepp (EX1007).....	55
(d) Sagman (EX1006)	58
(e) Robertson (EX1009).....	61
(f) Glaser (EX1019).....	64

TABLE OF CONTENTS
(continued)

	Page
B. Ground 1: Claims 1-11, 14, and 16-21 Are Obvious Over Shimizu, Fox, Benedict and Wolfe	66
1. Independent Claim 1	66
(a) “creating a show structure of nodes, each node identifying a resource from a plurality of accessible resources, at least some of which are network accessible resources;” (Claim 1[a]).....	72
(b) “without requiring user input, automatically accessing a plurality of said network accessible resources identified by a corresponding node; and” (Claim 1[b])	94
(c) “causing content from said network accessible resources corresponding to each of said accessed resources to be automatically accessed and presented through said user's computer in accordance with said show structure such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed,” (Claim 1[c]).....	100
(d) “wherein said step of creating further comprises the step of providing interactively variable duration information, representing the duration that said content corresponding to said resource is presented so as to enable a user to vary said duration.” (Claim 1[d])	109
2. Claim 2	115
3. Claims 3-6	116
4. Claims 7-9	117
5. Claim 10	118
6. Claim 11	121
7. Claims 14 and 16.....	121

TABLE OF CONTENTS

(continued)

	Page
8. Claim 17	124
9. Claim 18	125
10. Claim 19	126
11. Claim 20	126
12. Claim 21	127
C. Ground 2: Claims 22-29, 32-45 Are Obvious Over Shimizu, Fox, Benedict, Wolfe and Guinan.....	132
1. Independent Claim 22	132
(a) “receiving a request for a user defined online search from a remote user;” (Claim 22[a]).....	134
(b) “providing a plurality of search results and their corresponding resource locators in response to said request...” (Claim 22[b][1]	134
(c) “...automatically creating a show structure of nodes based on said search results, each node identifying a resource from a plurality of accessible resources via said communications network” (Claim 22[b][2]).....	142
(d) “without requiring user input automatically accessing a plurality of said resources;” (Claim 22[c]).....	144
(e) “retrieving content corresponding to each of said accessed resources; and” (Claim 22[d])	144
(f) “automatically accessing and presenting said retrieved content through said user’s computer in accordance with said show structure....” (Claim 22[e][1]).....	144
(g) “...such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.” (Claim 22[e][2]).....	145

TABLE OF CONTENTS

(continued)

	Page
2. Claim 23	146
3. Claims 24-25	146
4. Claim 26	147
5. Claim 27	147
6. Claim 28	148
7. Claim 29	150
8. Claim 32	152
9. Claims 33-35	153
10. Claims 36-37	154
11. Claims 38-40	155
12. Claims 41-42	156
13. Claims 43-44	158
14. Claim 45	158
D. Ground 3: Claims 46-62 Are Obvious Over Shimizu, Fox, Benedict, Wolfe, Guinan and Miller	160
1. Independent Claim 46	160
(a) “providing a predefined list of categories;” (Claim 46[a])	161
(b) “providing a plurality of search results based on user selections and their corresponding URLs in response to a user request related to a category among said predefined list of categories;” (Claim 46[b])	166
(c) “automatically creating a show structure of nodes based on said search results, each node identifying a resource from a plurality of accessible resources via said communications network, wherein said show structure is provided in response to said user request;” (Claim 46[c])	167

TABLE OF CONTENTS
(continued)

	Page
(d) “without requiring user input automatically accessing a plurality of said resources; and” (Claim 46[d])	168
(e) “automatically accessing and presenting content corresponding through said accessed resources to said user’s computer in accordance with said show structure such that during said presentation of at least said content from said accessed resources, at least some of said other network accessible resources are being accessed.” (Claim 46[e]).....	168
2. Claims 47-49	168
3. Claims 50-51	170
4. Claims 54-56	171
5. Claims 57-58	171
6. Claims 59-61	172
7. Claim 62	173
E. Ground 4: Claims 63-67 Are Obvious Over Shimizu, Fox, Benedict, Wolfe, and Sagman	174
1. Independent Claim 63	174
(a) “allowing the selection of at least one resource from a plurality of accessible resources via said communications network, wherein each of said resources is associated with a corresponding identifier displayed on a first section of said user’s display;” (Claim 63[a]).....	176
(b) “providing an interface for creating a show structure of nodes each node identifying a corresponding resource, by allowing generation of a list of said resources by dragging and dropping said identifiers on a second section of said user’s display;” (Claim 63[b]).....	182

TABLE OF CONTENTS

(continued)

	Page
(c) “without requiring user input, automatically accessing each of said resources identified by each of said nodes from said network accessible resources in accordance with said show structure;” (Claim 63[c]) (d) “automatically retrieving content corresponding to each of said accessed resources; and” (Claim 63[d]) (e) “causing the presentation of said retrieved content automatically accordance with said show structure such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.” (Claim 63[e]).....	193 193 193
2. Claim 64	195
3. Claim 65	196
4. Claim 66	200
5. Claim 67	202
F. Ground 5: Claims 11-16 Are Obvious Over Ground 1 Prior Art With Glaser.....	206
1. Claims 11-12	206
2. Claims 13-16	210
G. Ground 6: Claims 30-31 Are Obvious Over Ground 2 Prior Art with Filepp.....	214
H. Ground 7: Claims 52-53 Are Obvious Over Ground 3 Prior Art with Filepp.....	214
I. Ground 8: Claim 65 Is Obvious Over Ground 4 Prior Art with Robertson.....	219
VI. NO SECONDARY CONSIDERATIONS OF NONOBVIOUSNESS.....	223
VII. CONCLUSION.....	225

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

I, Christopher M. Schmandt, declare as follows:

I. INTRODUCTION AND QUALIFICATIONS

A. Qualifications and Experience

1. I have recently retired from my position as a Principal Research Scientist at the Media Laboratory at Massachusetts Institute of Technology (“M.I.T”), after 40 years of employment by M.I.T. In that role, I also served as faculty for the M.I.T. Media Arts and Sciences academic program. I have more than 40 years of experience in the field of Media Technology, and was a founder of the M.I.T. Media Laboratory.

2. I received my Bachelor of Science degree in Electrical Engineering and Computer Science from M.I.T. in 1978, and my Master of Science degree in Visual Studies (Computer Graphics), also from M.I.T. I had been employed at M.I.T. since 1980, initially at the Architecture Machine Group, which was an early computer graphics research lab. In 1985, I helped found the Media Laboratory and continued to work there until my retirement. I ran a research group titled “Living Mobile.” My research spanned distributed communication and collaborative systems, with an emphasis on multi-media and user interfaces; I have more than 70 published conference and journal papers and one book in these fields.

3. In my faculty position, I taught courses and directly supervised student

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

research and theses at the Bachelors, Masters, and Ph.D. level. I oversaw the Masters and Ph.D. thesis programs for the entire Media Arts and Sciences academic program. Based on the above experience and qualifications, I have a solid understanding of the knowledge and perspective of a person of ordinary skill in this technical field since at least 1980.

4. In the course of my nearly 40 years of research at MIT, I conceived of and helped implement, with my students, a number of hyper-media systems which are analogous to the invention claimed in the '932 patent. The earliest of these, Gaze Orchestrated Dynamics, dated from 1982, and provided an eye gaze driven user interface to browse a network of linked video snippets; this included managing and synchronizing a separate digital audio track for simultaneous presentation as a navigational cue.

5. I built a number of systems oriented toward voice access to information, often over a telephone connection. In these cases, the content consisted of text and/or audio material, with most voice user interfaces providing a serialized default path through the content. Consider a unified voice / text message system (Voiced Mail, 1982 and Phone Slave, 1983). A visual user interface to email typically presents headers for all the new messages simultaneously, and a user clicks on what they wish to hear. But when the content is audio, the interaction is by necessity

serialized, with the system providing a default path of reading or playing messages in some order. Of course in this situation the user maintains some control over the presentation, and can always change the time spent on each message by interrupting (the equivalent to clicking on a control button in a GUI). Additionally, as we learned that users quickly gain facility in understanding synthetic speech, each user could independently control the playback speed, and hence time on each message, of both the synthetic and recorded audio content.

6. Although any individual presentation in such systems follows a given path, the content nodes are in fact a network, and can be traversed along different paths. For example, a user may request only urgent messages, in which a new path is calculated to present those messages first, or messages from a particular sender, or a particular date.

7. Another example of a complex network of nodes was non-real-time direction giving (Direction Assistance, 1987) in which the content nodes were literally each street intersection in Cambridge, MA, with the content consisting of instructions for a driver. This system was built so a user could take notes while listening, before starting to drive, and as we learned that people take notes at very different speeds, voice control of the pause between each piece of directions was added, for flexible flow. Systems such as these also exhibit search, in that the user

specifies a destination address, and the system calculates a route (i.e. a path through the intersections) for presentation. We also incorporated much of the same logic into one of the earliest real-time spoken driving direction systems (Back Seat Driver, 1988) in which temporal presentation takes on the added burden of calculating the proper time to speak based on driving speed, and sometimes needs to merge two direction turns into a single presentation to give adequate warning.

8. Hyperspeech (1993) used a more conventional hypertext model, only each content node was an audio recording by a particular speaker. The recordings were collected automatically by the system phoning the talent, asking questions, and recording answers. A user's path through the linked content recordings was then chosen according to a question, or series of questions, a Hyperspeech caller would ask. I also worked with the Unix cron facility, which is used for scheduling tasks to occur at specified times. I also extended the crontab file functionality to support recurring events, and built speech and graphical user interfaces around the resulting extended crontab functionality (Caltalk, 1990).

9. The above are representative examples of projects from my research group which involved hypertext-like systems analogous to the '932 patent for navigating through a set of information using one or more paths.

10. My Curriculum Vitae is attached as **Exhibit A**.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

11. I have been retained by counsel for Petitioners to provide my expert opinion in connection with the above-captioned proceeding as set forth herein.

B. Materials Considered

12. The analysis that I provide in this Declaration is based on my education, research, and experience, as well as the documents I have considered. In forming my opinions, I have read and considered U.S. Patent No. 8,965,932 B2 (“’932 patent”) [Ex. 1001] and its prosecution history. I have cited to the following documents in my analysis below:

Exhibit No.	Description of Document
1001	U.S. Patent No. 8,965,932 B2 to Damon Torres (filed Dec. 21, 2005, issued Feb. 24, 2015) (“’932” or “’932 patent”)
1003	U.S. Patent No. 5,634,062 to Takeshi Shimizu et al. (filed October 20, 1994, issued May 27, 1997) (“Shimizu”)
1004	Excerpts from Robert C. Benedict, Jr., <i>Using Harvard Graphics 2 for Windows</i> (1993) (“Benedict”)
1005	Excerpts from David Fox & Troy Downing, <i>HTML Web Publisher’s Construction Kit</i> (1995) (“Fox”)
1006	Excerpts from Stephen W. Sagman, <i>Running Microsoft PowerPoint 4 for Windows</i> (1994) (“Sagman”)

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

Exhibit No.	Description of Document
1007	U.S. Patent No. 5,347,632 to Robert Filepp et al. (filed July 28, 1989, issued September 13, 1994) (“Filepp”)
1008	Catherine Guinan et al., <i>Information Retrieval from Hypertext Using Dynamically Planned Guided Tours</i> , Proceedings of the ACM Conference on Hypertext (Nov. 1992), pp.122-130 (“Guinan”)
1009	U.S. Patent No. 6,486,895 B1 to George G. Robertson et al. (filed September 8, 1995, issued November 26, 2002) (“Robertson”)
1010	Excerpts from Michael Miller, <i>Using CompuServe</i> (2d ed. 1994) (“Miller”)
1014	U.S. Patent No. 5,715,445 to Mark A. Wolfe (filed June 5, 1995, issued February 3, 1998) (“Wolfe”)
1016	U.S. Patent No. 5,217,119 to Dale Hollingsworth (filed June 4, 1992, issued June 8, 1993) (“Hollingsworth”)
1017	U.S. Patent No. 5,884,056 to David A. Steele et al. (filed December 28, 1995, issued March 16, 1999) (“Steele”)
1019	U.S. Patent No. 5,793,980 to Robert D. Glaser et al. (filed November 30, 1994, issued August 11, 1998) (“Glaser”)
1020	Excerpts from <i>Microsoft Press Computer Dictionary</i> (2d ed. 1994)
1023	Claim Construction Order issued in <i>Robocast, Inc. v. Microsoft Corp.</i> , No. 1:10-cv-01055 RGA (D. Del.) and <i>Robocast, Inc. v. Apple Inc.</i> , No. 1:11-cv 00235 RGA (D. Del.), dated June 28, 2013
1024	Prosecution history of U.S. Patent No. 7,155,451 B1

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

Exhibit No.	Description of Document
1025	Prosecution history of U.S. Patent No. 8,606,819 B2
1026	Prosecution history of U.S. Patent No. 8,965,932 B2
1027	Prosecution history for U.S. Patent Appl. Serial No. 08/922,063
1028	Prosecution history for U.S. Provisional Patent Appl. Serial No. 60/025,360

II. PERSON OF ORDINARY SKILL IN THE ART

13. I understand that, under the patent laws in effect before the America Invents Act (“AIA”) of 2011, an assessment of claims of a patent filed before the AIA took effect should be undertaken from the perspective of a person of ordinary skill in the art as of the earliest claimed priority date (i.e., the “time the invention was made”). I have preliminarily for purposes of my Declaration only assumed that date to be September 3, 1996, the filing date of the earliest application to which the ’932 patent claims priority (provisional application no. 60/025,360).¹

¹ I express no opinion on whether the ’932 patent is actually entitled to the benefit of the provisional application to which it claims priority (60/025,360). This issue is not relevant here because all of the prior art references relied upon predate that date. In the event entitlement to the provisional filing date may later become relevant, for example based on arguments later presented by Patent Owner, I reserve my right to provide opinions on that issue at that time.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

14. I have also been advised that to determine the appropriate level of a person having ordinary skill in the art, the following factors may be considered: (1) the types of problems encountered by those working in the field and prior art solutions thereto; (2) the sophistication of the technology in question, and the rapidity with which innovations occur in the field; (3) the educational level of active workers in the field; and (4) the educational level of the inventor.

15. The '932 patent states that it generally relates to “computer networks, and specifically to a method of sequencing and scheduling web pages, via a suite of software applications.” ('932, 1:22-24.)

16. In my opinion, a person of ordinary skill in the art as of September 1996 would have possessed a bachelor's degree in electrical engineering, computer science, or similar field, with at least two years of experience in developing and implementing network-based computer systems, such as systems for storing and retrieving information over the Internet or communicating using the Web. A person could also have qualified as a person of ordinary skill in the art with some combination of (1) more formal education (such as a master's of science degree) and less technical experience, or (2) less formal education and more technical or professional experience. I am also informed that, in an earlier *inter partes* review proceeding involving the '932 patent, Patent Owner proposed the following

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

articulation of the level of ordinary skill in the art:

A person of ordinary skill in the art at the time of the invention would have the equivalent of a four-year degree from an accredited institution (usually denoted as a B.S. degree) in computer science, computer engineering or the equivalent, and experience with, or exposure to, topics in computer networking, multimedia, and web systems. A person of ordinary skill in the art would also have approximately 3-5 years of professional experience with these topics. Additional graduate education could substitute for professional experience, while significant experience in the field might substitute for formal education.

For purposes of my analysis, Patent Owner's proposed articulation is not meaningfully different from mine, and therefore my analysis, opinions and conclusions in my Declaration would not change if Patent Owner's articulation were applied.

17. My opinions regarding the level of ordinary skill in the art are based on, among other things, my over 40 years of experience in the field of computer science, my understanding of the basic qualifications that would be relevant to an engineer or scientist tasked with investigating methods and systems in the relevant area, and my familiarity with the backgrounds of colleagues, co-workers, and employees, both past and present.

18. Although my qualifications and experience exceed those of the

hypothetical person having ordinary skill in the art defined above, my analysis and opinions regarding the '932 patent have been based on the perspective of a person of ordinary skill in the art as of September 1996.

III. STATEMENT OF LEGAL PRINCIPLES

A. Claim Construction

19. I understand that a purpose of claim construction is to determine what a person of ordinary skill in the art would have understood the claim terms to mean. Claim terms are generally given their ordinary and customary meaning, which is the meaning that the term would have to a person of ordinary skill in the art in question as of the effective filing date.

20. I understand that the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification. I understand that the patent specification, under the legal principles, has been described as the single best guide to the meaning of a claim term, and is thus highly relevant to the interpretation of claim terms. And I understand for terms that do not have a customary meaning within the art, the specification usually supplies the best context of understanding the meaning of those terms.

21. I further understand that other claims of the patent in question, both

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

asserted and unasserted, can be valuable sources of information as to the meaning of a claim term. Because the claim terms are normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims. Differences among claims can also be a useful guide in understanding the meaning of particular claim terms.

22. I understand that the prosecution history can further inform the meaning of the claim language by demonstrating how the inventors understood the invention and whether the inventors limited the invention in the course of prosecution, making the claim scope narrower than it otherwise would be. Extrinsic evidence, such as dictionaries, may also be consulted in construing the claim terms.

23. I understand that, in *Inter Partes* Review (IPR) proceedings, a claim of a patent shall be construed using the same claim construction standard that would be used to construe the claim in a civil action filed in a U.S. district court (which I understand is called the “*Phillips*” claim construction standard), including construing the claim in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.

24. I have been instructed by counsel to apply the “*Phillips*” claim construction standard for purposes of interpreting the claims in this proceeding, to

the extent they require an explicit construction. The description of the legal principles set forth above thus provides my understanding of the “*Phillips*” standard as provided to me by counsel.

25. I understand that certain terms appearing in the challenged claims have been the subject of district court claim constructions in previous litigations (involving related U.S. Patent No. 7,155,451 B1 (“’451 patent”), which shares a near-verbatim written description with the ’932 patent) involving Apple Inc. and Microsoft Corporation. I have reviewed a document called “**Claim Construction**,” dated June 28, 2013, which states that it is a claim construction order from the District of Delaware issued in *Robocast, Inc. v. Microsoft Corp.*, No. 10-1055 RGA and *Robocast, Inc. v. Apple Inc.*, No. 11-235 RGA, which provides constructions for various terms recited in the challenged patents. I will identify those constructions in my analysis in **Part V** below. For purposes of my analysis here, I do not believe express claim constructions are necessary because the prior art renders the claims obvious under the claim constructions adopted by the district court or any other reasonable construction.

B. Obviousness

26. I understand that a patent claim is obvious if, as of the effective filing date, it would have been obvious to a person having ordinary skill in the field of the

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

technology (the “art”) to which the claimed subject matter belongs.

27. I understand that the following factors should be considered in analyzing obviousness: (1) the scope and content of the prior art; (2) the differences between the prior art and the claims; and (3) the level of ordinary skill in the pertinent art. I also understand that certain other facts known as “secondary considerations” such as commercial success, unexplained results, long felt but unsolved need, industry acclaim, simultaneous invention, copying by others, skepticism by experts in the field, and failure of others may be utilized as indicia of nonobviousness. I understand, however, that secondary considerations should be connected, or have a “nexus,” with the invention claimed in the patent at issue.

28. I understand that a reference qualifies as prior art for obviousness purposes when it is analogous to the claimed invention. The test for determining what art is analogous is: (1) whether the art is from the same field of endeavor, regardless of the problem addressed, and (2) if the reference is not within the field of the inventor’s endeavor, whether the reference still is reasonably pertinent to the particular problem with which the inventor is involved.

29. I understand that a person of ordinary skill in the art is assumed to have knowledge of all prior art. I understand that one skilled in the art can combine various prior art references based on the teachings of those prior art references, the

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

general knowledge present in the art, or common sense. I understand that a motivation to combine references may be implicit in the prior art, and there is no requirement that there be an actual or explicit teaching to combine two references. Thus, one may take into account the inferences and creative steps that a person of ordinary skill in the art would employ to combine the known elements in the prior art in the manner claimed by the patent at issue. I understand that one should avoid “hindsight bias” and ex post reasoning in performing an obviousness analysis. But this does not mean that a person of ordinary skill in the art for purposes of the obviousness inquiry does not have recourse to common sense.

30. I understand that when determining whether a patent claim is obvious in light of the prior art, neither the particular motivation for the patent nor the stated purpose of the patentee is controlling. The primary inquiry has to do with the objective reach of the claims, and that if those claims extend to something that is obvious, then the entire patent claim is invalid.

31. I understand one way that a patent can be found obvious is if there existed at the time of the invention a known problem for which there was an obvious solution encompassed by the patent’s claims. I understand that a motivation to combine various prior art references to solve a particular problem may come from a variety of sources, including market demand or scientific literature. I understand

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

that a need or problem known in the field at the time of the invention can also provide a reason to combine prior art references and render a patent claim invalid for obviousness. I understand that familiar items may have obvious uses beyond their primary purpose, and that a person of ordinary skill in the art will be able to fit the teachings of multiple prior art references together like the pieces of a puzzle. I understand that a person of ordinary skill is also a person of at least ordinary creativity. I understand when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this finite number of predictable solutions leads to the anticipated success, I understand that the invention is likely the product of ordinary skill and common sense, and not of any sort of innovation. I understand that the fact that a combination was obvious to try might also show that it was obvious, and hence invalid, under the patent laws. I understand that if a patent claims a combination of familiar elements according to known methods, the combination is likely to be obvious when it does not more than yield predictable results. Thus, if a person of ordinary skill in the art can implement a predictable variation, an invention is likely obvious. I understand that combining embodiments disclosed near each other in a prior art reference would not ordinarily require a leap of inventiveness.

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

32. I understand that obviousness may be shown by demonstrating that it would have been obvious to modify what is taught in a single piece of prior art to create the patented invention. Obviousness may also be shown by demonstrating that it would have been obvious to combine the teachings of more than one item of prior art. I understand that a claimed invention may be obvious if some teaching, suggestion, or motivation exists that would have led a person of ordinary skill in the art to combine the invalidating references. I also understand that this suggestion or motivation may come from the knowledge of a person having ordinary skill in the art, or from sources such as explicit statements in the prior art. I understand that when there is a design need or market pressure, and there are a finite number of predictable solutions, a person of ordinary skill may be motivated to apply common sense and his skill to combine the known options in order to solve the problem.

33. I understand the following are examples of approaches and rationales that may be considered in determining whether a piece of prior art could have been combined with other prior art or with other information within the knowledge of a person having ordinary skill in the art:

- (1) Some teaching, motivation, or suggestion in the prior art that would have led a person of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention;

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

- (2) Known work in one field of endeavor may prompt variations of it for use in the same field or a different field based on design incentives or other market forces if the variations would have been predictable to a person of ordinary skill in the art;
- (3) Combining prior art elements according to known methods to yield predictable results;
- (4) Applying a known technique to a known device, method, or product ready for improvement to yield predictable results;
- (5) Applying a technique or approach that would have been “obvious to try” (choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success);
- (6) Simple substitution of one known element for another to obtain predictable results; or
- (7) Use of a known technique to improve similar products, devices, or methods in the same way.

34. I understand that, when determining whether a claimed combination is obvious, the correct analysis is not whether one of ordinary skill in the art, writing on a blank slate, would have chosen the particular combination of elements described in the claim. Instead, I understand the correct analysis considers whether

one of ordinary skill, facing the wide range of needs created by developments in the field of endeavor, would have seen a benefit to selecting the combination claimed.

35. I understand that the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference. The test for obviousness, in other words, is not whether the references could be physically combined but whether the claimed inventions are rendered obvious by the teachings of the prior art as a whole.

IV. THE '932 PATENT

A. Overview of the Specification

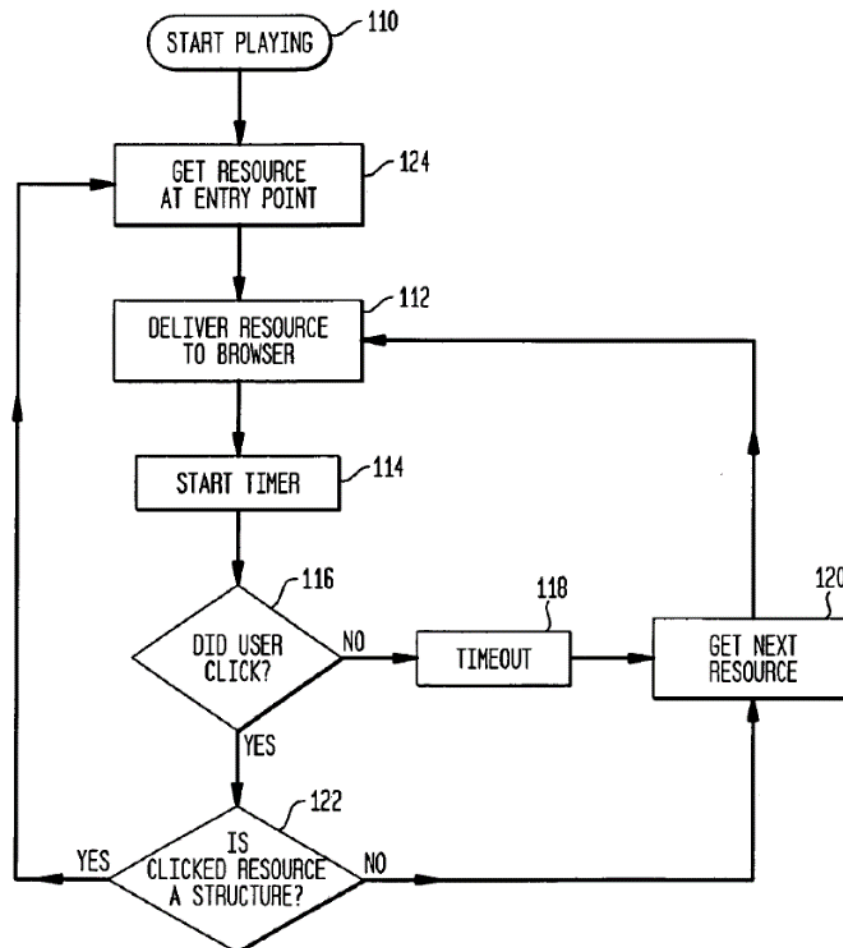
36. The '932 patent, entitled "Automated Content Scheduler and Displayer," states that it "relates generally to computer networks, and specifically to a method of sequencing and scheduling web pages, via a suite of software applications." ('932, 1:22-24.)

37. The '932 patent acknowledges that the Internet was already well-known and had "grown tremendously" with the popularization of the World Wide Web. ('932, 1:28-33.) The patent explains that various technology standards relating to the Web were popularized including Hypertext Transfer Protocol ("HTTP"), Hypertext Markup Language ("HTML") and Uniform Resource Locators ("URLs"). ('932, 1:40-43.) Using these and other technologies, the patent states

that web browser software provided an environment that “proved to be very popular” and led to the web’s “mass adoption.” (’932, 1:62-65.) The patent further acknowledges known web technologies for a user to automatically receive personalized information updates both by selecting content sources from a list and by using collected user interest profiles. (’932, 2:16-21, 2:29-35.)

38. The ’932 patent describes methods for a user to be automatically presented with content of a series of resources, such as web pages, using what the patent calls a “show structure.” (’932, 2:59-63.) A “show structure” contains a set of nodes, each node “represent[ing] a resource such as a web page, an executable file, or a data file stored at a predetermined location and accessed via its corresponding address.” (’932, 2:61-65.) Each node includes both an address from which the content may be accessed and an indication of the duration for which the content will be presented. (’932, 3:30-35.) A show structure defines one or more paths through the set of nodes. (’932, 3:10-11.) Figure 4 of the ’932 patent, reproduced below, outlines how a user may either “on-the-fly” interactively navigate the arrangement of resources specified by the show structure (*e.g.*, steps 116 and 122) or, much like traditional broadcast television or an automatic slideshow, “simply watch[] a series of pre-edited resources” (*e.g.*, steps 114-118):

FIG. 4
RoboSurf



(’932, Fig. 4; *see also id.*, 3:14-22, 11:5-36.) The ’932 patent explains that a show structure itself can be generated in a number of ways, including manually by a user collecting URLs, or automatically in connection with providing results to a user’s search query. (’932, 8:24-28, 11:61-12:3.) I discuss additional aspects of the ’932 patent in my analysis of the claims below.

B. The Challenged Claims

39. This Declaration addresses claims 1-67 of the '932 patent. Independent claim 1 is representative and recites:

1. A method for providing content via a communications network on a user's computer, obtained from a plurality of resources in an organized arrangement comprising the steps of:
 - [a] creating a show structure of nodes, each node identifying a resource from a plurality of accessible resources, at least some of which are network accessible resources;
 - [b] without requiring user input, automatically accessing a plurality of said network accessible resources identified by a corresponding node; and
 - [c] causing content from said network accessible resources corresponding to each of said accessed resources to be automatically accessed and presented through said user's computer in accordance with said show structure such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed,
 - [d] wherein said step of creating further comprises the step of providing interactively variable duration information, representing the duration that said content corresponding to said resource is presented so as to enable a user to vary said duration.

(’932, 17:42-64 (Claim 1; bracketed notation (e.g., **[a]**) added).)

40. I address the claims further in my detailed analysis in **Part V** below.

V. APPLICATION OF THE PRIOR ART TO CHALLENGED CLAIMS

41. I have reviewed and analyzed the prior art references and materials listed in **Part I.B** above. In my opinion the claims of the ’932 patent would have been obvious to a person of ordinary skill in the art based on the following combinations of the prior art.

Ground	Claims	Basis for Challenge Under § 103
1	1-11, 14, 16-21	Shimizu, Benedict, Fox, Wolfe
2	22-29, 32-45	Shimizu, Benedict, Fox, Wolfe, Guinan
3	46-51, 54-62	Shimizu, Benedict, Fox, Wolfe, Guinan, Miller
4	63-67	Shimizu, Benedict, Fox, Wolfe, Sagman
5	11-16	Ground 1 Prior Art with Glaser
6	30-31	Ground 2 Prior Art with Filepp
7	52-53	Ground 3 Prior Art with Filepp
8	65	Ground 4 Prior Art with Robertson

42. I am informed by counsel that each of the references cited in the grounds above qualifies as prior art to the challenged claims because each reference was filed and/or published before the ’932 patent priority date (September 3, 1996).

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

I also understand that the grounds listed above may be numbered differently from the numbering in the IPR petitions filed by Petitioner, as I am informed that they may be split among multiple petitions.

A. Brief Summary of Prior Art

1. References Common to All Grounds

(a) Shimizu (EX1003)

43. **Shimizu**, U.S. Patent No. 6,534,062, is entitled “System for Managing Hypertext Node Information and Link Information.” I am informed that Shimizu qualifies as prior art because it was filed on October 20, 1994, which is before the earliest priority date of the challenged claims.

44. As its title indicates, Shimizu discloses a system for managing information about nodes and links in a “hypertext” system. “Hypertext” generally refers to a way of organizing and presenting information in which information is presented with links or references (commonly referred to as “hyperlinks”) to other information. Shimizu explains that “[h]ypertext apparatuses were first developed as a system capable of holding information in units called nodes, of displaying character information held by the nodes, and of linking a plurality of nodes as desired.” (Shimizu, 1:17-20.) As Shimizu observes, although hypertext systems could present textual information, they also supported many other types of

information such as images, sounds, and motion pictures. (Shimizu, 1:23-29.)

45. Hypertext systems were well-known since at least the 1980s and, by the priority date, were exceedingly popular. An early example of a well-known and commercially available hypertext-based system was “HyperCard” from Apple Computer, Inc., which allowed users to organize information into virtual “stacks” of “cards,” which could be hyperlinked. Another example of a “hypertext” system common by the mid-1990s – and no doubt the most widely known example by the priority date – was the World Wide Web. The Web provides an example of a hypertext system because web pages can present text, graphics, and/or other types of information and can include hyperlinks to other pages. (Fox, EX1005, pp.8-10 (describing the Web as an example of a hypertext system).)

46. Figure 1 of Shimizu depicts an exemplary hypertext system with a number of nodes and links:

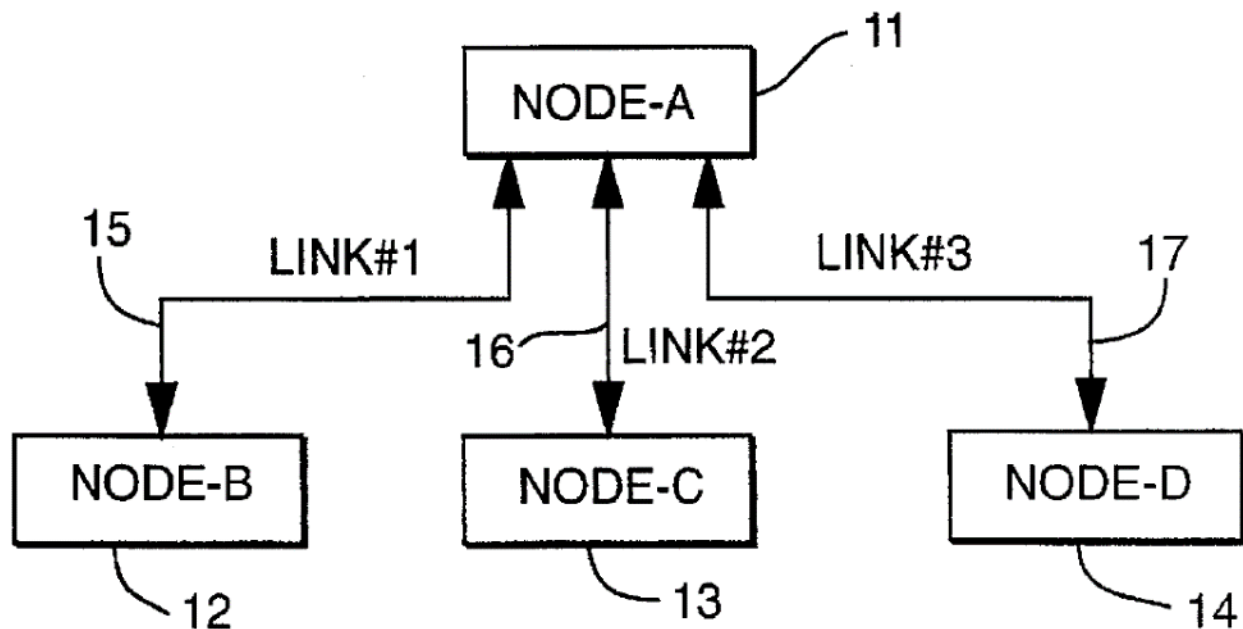


FIG. 1

(Shimizu, Fig. 1.) The hypertext arrangement depicted in Figure 1 shows Node-A through Node-D (11, 12, 13, and 14) and links 15, 16, and 17. Node-A 11 has three links 15, 16, and 17 to Node-B, Node-C, and Node-D, respectively. (Shimizu, 1:56-60.) “Link information is held by each node.” (Shimizu, 1:61.) “In the example above, the access from node A 11 is made first to node B 12 then to node C 13 and to node D 14, in the order in which the first link information 21 (link #1), second link information 22 (link #2), and third link information 23 (link #3) are read out consecutively.” (Shimizu, 2:21-26.)

47. Shimizu focuses on techniques for managing and keeping track, in a

hypertext system, of (a) node information and (b) link information. (Shimizu, 1:8-15.) Node information can include “not only characters but also monochromatic and color images, sounds and motion pictures.” (Shimizu, 1:23-28.) Two key components of Shimizu, for purposes of my analysis, are node information management part 5 and link information management part 6 shown in Figure 10:

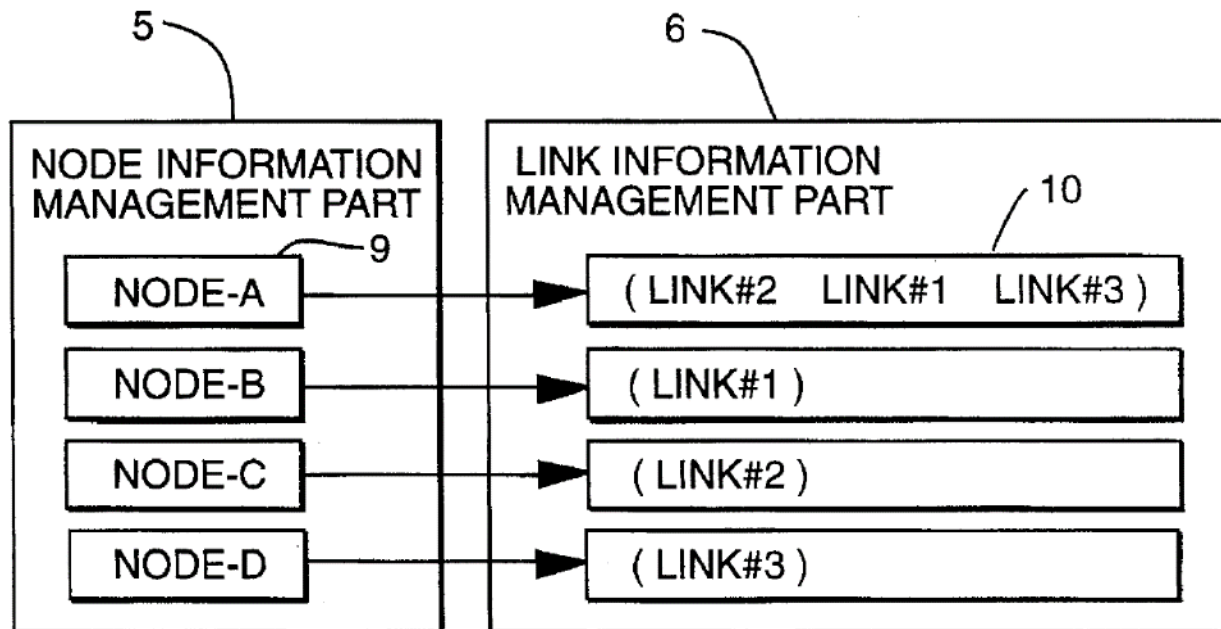


FIG. 10

(Shimizu, Fig. 10.) Node information management part 5, shown on the left above, “manages a plurality of node information structures holding the information about each node.” (Shimizu, 6:31-33.) Node information management part 5 manages node information structures for each node, each structure specifying information about the node. Figure 6 shows an exemplary node information structure 9:

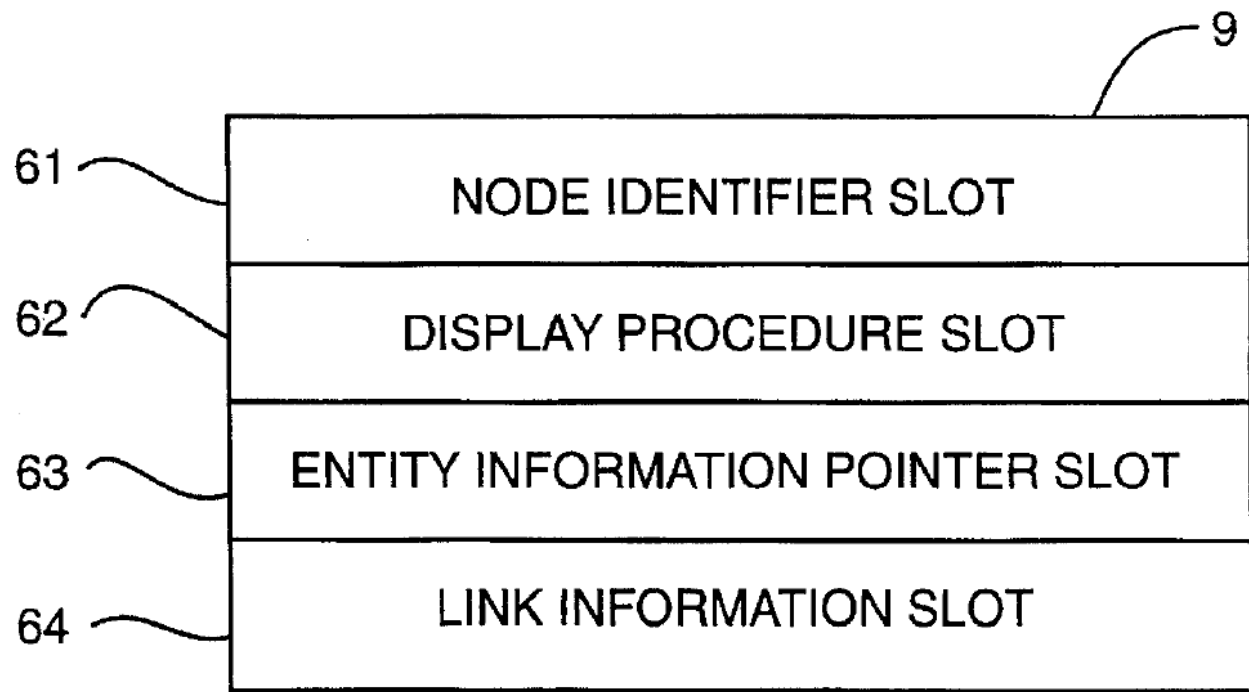


FIG. 6

(Shimizu, Fig. 6.) Node information structure 9 holds a number of members including, as shown, node identifier slot 61, display procedure slot 62, entity information pointer slot 63, and link information slow 64. As Shimizu explains:

The node identifier slot 61 is used to identify uniquely this node information structure within the system. The display procedure slot 62 retains the pointer pointing to a procedure program function used when the node in question is to be displayed on the screen. The entity information pointer slot 63 retains the pointer pointing to the entity information to be displayed by the node in question. The entity information thus pointed to may be a character string, image

information, motion pictures or sounds. The link information slot **64** retains, in a list format data constitution, the pointer pointing to the link information structure for holding the link information.

(Shimizu, 6:50-62.)

48. Figure 10 above also shows, on the right, link information management part **6**, which “manages a plurality of link information structures defining the relations between the nodes.” (Shimizu, 6:33-35.) As its name implies, the link information structure specifies information about a link associated with a node. Figure 7 shows a “link information structure managed by the link information management part **6**.”

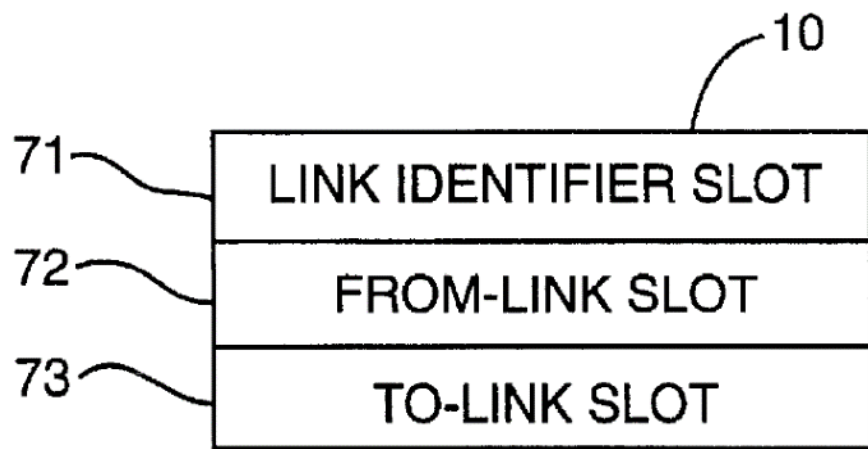


FIG. 7

(Shimizu, Fig. 7.) Link information structure **10** as shown in Figure 7 includes a

link identifier slot **71**, a from-link slot **72**, and a to-link slot **73**. Shimizu explains:

The link identifier slot **71** is used to identify uniquely this link information structure within the system. The from-link slot **72** retains the pointer pointing to the node information structure of, say, the node A when a link is set illustratively to link the node A to the node B. The to-link slot **73** retains the pointer pointing to the node information structure of the node B when, as in the example above, a link is set to link the node A to the node B.

(Shimizu, 7:1-9.)

49. Shimizu goes on to explain how the system uses the node and link information structures discussed above to present the nodes to the user in a particular order as indicated by the link information. Figure 8 shows a user interface screen for presenting node content to the user:

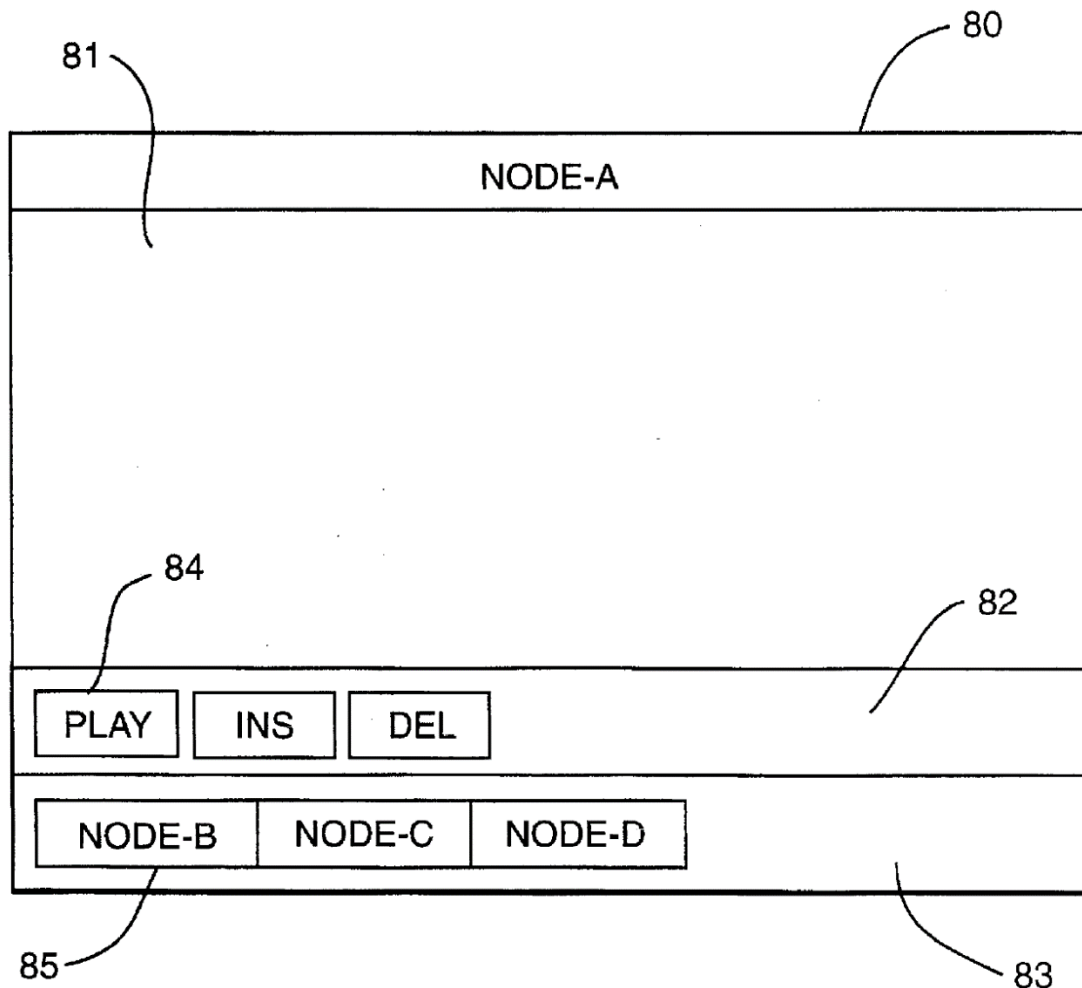


FIG. 8

(Shimizu, Fig. 8, 7:12-14 (“FIG. 8 is a view of a typical user interface screen display provided by an application program running on the embodiment.”).) When the user presses PLAY in Figure 8 (one of buttons **84**), the content associated with the current node (in this case NODE-A) is displayed in area **81** of window **80**. (Shimizu, 8:11-25, 8:49-59.) Figure 8 also includes display area **83** that “displays, in a left-justified manner and according to the link information order, icons **85** representing each of

the nodes linked starting from the current node.” (Shimizu, 7:30-33.) The node content displayed to the user, as mentioned, can include textual character strings, images, sounds, and motion pictures. (Shimizu, 8:52-59.)

50. Shimizu further discloses that node content is accessed and presented on the user’s computer in a continuous sequential fashion, one node after another, without requiring any input from the user beyond an initial click of the PLAY button. “The contents of the linked nodes are displayed in the order set by the link information.” (Shimizu, 8:15-16; *see also id.*, 8:67-9:4 (“In step **105** [of Figure 11], the next node information is reproduced in accordance with the next link information. The processing above is carried out so that the node C Is displayed followed by node B and node D, in accordance with the order of the link information.”).)² As Shimizu further explains, nodes can be displayed and switching to the next node can occur at predetermined intervals:

As described, the hypertext apparatus of the invention offers a user interface capable of putting into a desired order the links between the nodes to be managed and of manipulating the established link order. This makes it easy to create illustratively an application program

² Unless otherwise noted, all underlining appearing in quotes from prior art references was added for emphasis.

allowing the display of node contents to be switched in a desired order
and at predetermined intervals on the hypertext apparatus.

(Shimizu, 9:60-66.) I will provide more information about Shimizu in my discussion
of the claim limitations below.

(b) Benedict (EX1006)

51. **Benedict** is a book entitled *Using Harvard Graphics Version 2 for Windows*. I am informed that Benedict qualifies as prior art because it was published before the earliest priority date of the challenged claims.

52. Benedict describes a software program known as Harvard Graphics that provided functionality similar, in some respects, to well-known business presentation programs such as Microsoft PowerPoint. (Benedict, p.1 (“Harvard Graphics for Windows is a program designed for creating business presentations.”).) Harvard Graphics allowed users to create a “presentation” that included a series of slides. (Benedict, p.24 (“A *presentation* is a collection of slides stored in computer memory in a file on your hard disk.”) (italics in original).)

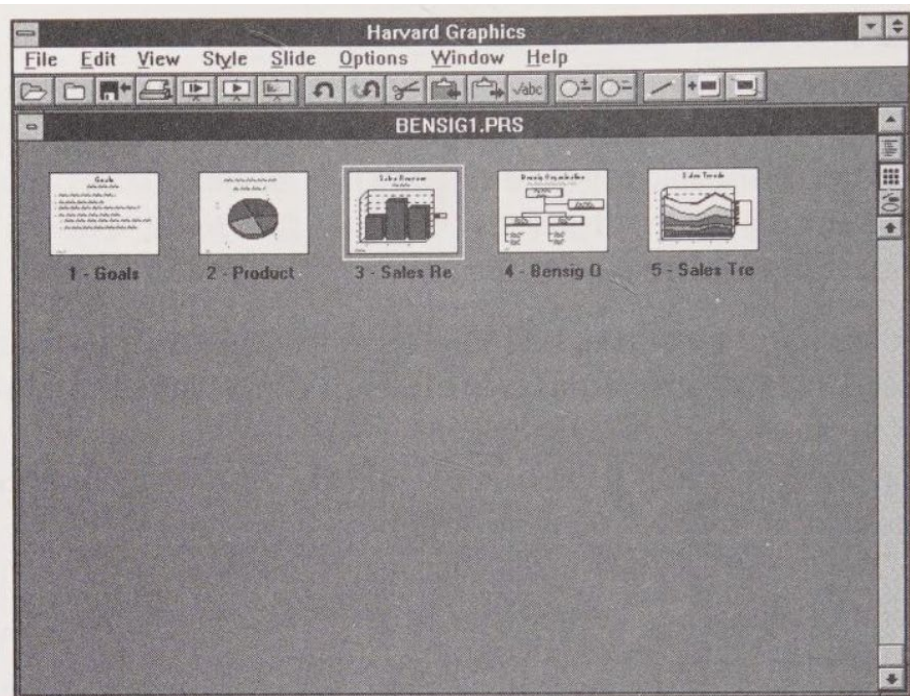
53. Benedict also discloses features for viewing a presentation (referred to as a “ScreenShow”), for example, by displaying slides in a specified sequential order. “As you create and enhance presentations, you can use the ScreenShow feature to evaluate your work. ScreenShows are on-screen presentations in which

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

the program displays each slide in order.” (Benedict, p.19.) Benedict further discloses a feature known as “HyperShows” in which slides in a ScreenShow can include hypertext-like links allowing the user to jump to other slides in the ScreenShow via on-screen hyperlinks. (Benedict, pp.442-453.)

54. Benedict describes a number of features for creating presentations, including adding new slides and specifying their content. (Benedict, pp.32-42.) Benedict also describes a “Slide Sorter” feature that displays icons for each slide in the presentation and allows the user to manipulate the order in which they will appear in the presentation:

Fig. 2.18
The Slide Sorter
window.



(Benedict, p.44, Fig. 2.18; *see also id.*, p.43 (“Figure 2.18 shows the Slide Sorter

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

displaying five slides. The Slide Sorter represents these slides as scaled down images.”).) The exemplary presentation above in Figure 2.18 includes five numbered and ordered slides (e.g., “1 - Goals,” “2 - Product,” “3 - Sales Revenue,” etc.). The user can reorder slides in a presentation by simply clicking on a slide with its mouse and dragging the slide to a new position. (Benedict, p.44.) “When you release the mouse button, the slide appears in its new location, and the other slides adjust to make room for the relocated slide.” (Benedict, p.44.)

55. Benedict discloses additional features (known as “ScreenShow Effects”) that allow the user to customize the display of the presentation. (Benedict, pp.438-442.) As shown in Figure 17.1 of Benedict, one of these ScreenShow Effects features allows the user to specify how long a particular slide in a presentation will be displayed before moving to the next slide:

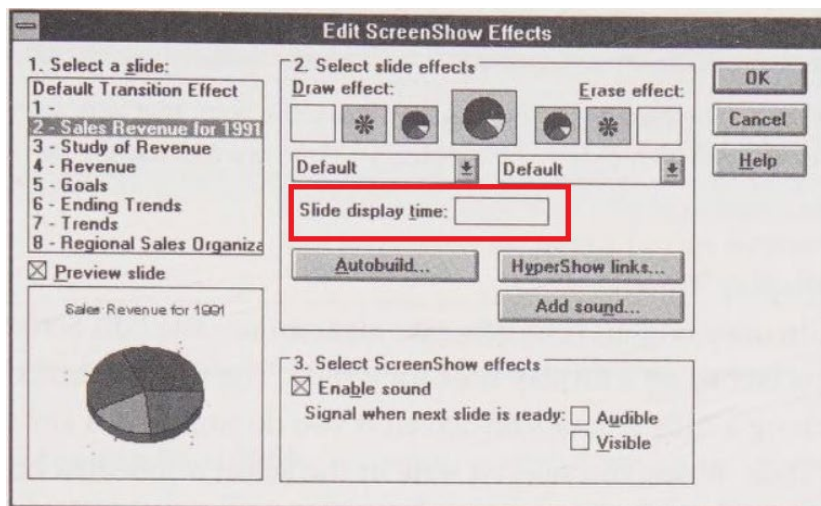


Fig. 17.1
The Edit
ScreenShow
Effects dialog box.

(Benedict, p.439, Fig. 17.1 (red box added).) The portion in the red box, as shown

above, allows the user to specify how long the selected slide will be displayed:

The display time determines how long a slide remains on-screen if you do not press a key to move to the next slide...

The default for the time is blank because Harvard Graphics continues to display the slide until you press a key. You can set a display time, however, by typing a time in the Slide display time: text box. You can enter the time in seconds or in minutes and seconds separated by a colon. Entering 23, for example, sets the display time to 23 seconds; entering 1:30 sets the display time to 1 minute and 30 seconds.

(Benedict, p.440 (*italics in original*).) Benedict explains that, using this feature, the user can create “self-running” presentations in which slides advance automatically after a specified period, without requiring user action:

Creating Self-Running Presentations

Self-running presentations automate your ScreenShow by displaying the slides for a specified length of time before the next slide appears—pressing a key is not necessary. At the end of the presentation, a HyperShow link causes the ScreenShow to start over from the beginning. The presentation continues until you press the Escape key. To create a self-running presentation, you must set the display time for the slides and create the link that begins the presentation again.

(Benedict, p.453 (**boldface in original**).)

56. I will provide more details about Benedict in my discussion of the claim

limitations below.

(c) Fox (EX1005)

57. **Fox** is a book entitled *HTML Web Publisher's Construction Kit*. I am informed that Fox qualifies as prior art because it was published before the earliest priority date of the challenged claims.

58. Fox covers a number of topics relating to the World Wide Web and how it works, including how to navigate the Web using Internet web browsers and how to create customized websites. (Fox, pp.xxiv-xxv.) Fox explains that, as is well-known today, “[a] Web page can be anything you imagine it to be: a hypertext document, graphics, sound, animation, tabular data, software files, a discussion group, a search index, a specialized program, and more.” (Fox, p.8; *see also id.*, p.9 (“The World Wide Web uses hypertext to link all its pages together.”).) Fox explains, as was well-known to ordinarily skilled artisans, that each resource on the web is identified by a Uniform Resource Locator (URL). (Fox, p.13; *see also id.*, pp.326-327 (describing IMG tag for identifying an image file).)

59. My Declaration relies on Fox primarily to demonstrate that it would have been obvious to implement the hypertext system described in Shimizu, along with the additional features of Benedict upon which I rely in this Declaration, in the form of a web-based hypertext application. For example, Fox confirms that web-

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

based techniques can be used to create sequential presentations akin to a slide show, similar to Shimizu and Harvard Graphics. For example, Fox discloses a technique in which a web page, after a specified amount of time, can cause the web browser to “refresh” and load another web page, which can be used to create slide show-type presentations of web content:

To load up a different document after a set amount of seconds, simply specify a full URL using the URL attribute. For example, to load the page at *http://www.smartpants.edu/sample.html*, after a minute, use the following markup as the first line of your HTML document:

```
<META HTTP-EQUIV="Refresh" CONTENT=60  
URL= http://www.smartpants.edu/sample.html>
```

You can use this trick to flip between two documents, animate a sequence of images, or create an automatic slide show of many Web pages.

(Fox, p.316 (italics in original; underlining added).) The concept of “refreshing” a web page was well-known to those familiar with web browsers, and was a common feature of many websites. Refreshing was commonly used to load an updated version of the *currently displayed* web page, but as shown above, the <META> tag used for this feature can specify the URL of any web page, including a different web page. In the example from Fox above, the <META> tag tells the web browser to “refresh” after 60 seconds by switching to a different web page located at

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

<http://www.smartpants.edu/sample.html>. I will provide more detail about Fox in my discussion of the claim limitations below.

(d) Wolfe (EX1014)

60. **Wolfe**, U.S. Patent No. 5,715,445, is entitled “Document Retrieval System Employing a Preloading Procedure.” I am informed that Wolfe qualifies as prior art because it states on its face that it was filed on June 7, 1995 (claiming priority as a continuation to an application filed September 2, 1994), before the earliest priority date of the challenged claims.

61. My Declaration relies on Wolfe for limitations of the challenged independent claims reciting the ability to access resources while other resources are being presented. Claim 1, for example, recites “**during said presentation of at least Some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.**”

62. This claimed feature recites nothing more than a well-known technique, commonly referred to as “pre-fetching” or “pre-loading,” in which a computer fetches or loads data *before* the point in time when the data is actually needed. When the computer *later* needs that requested data, it can utilize it immediately because it has already been fetched or loaded.

63. Pre-loading was a commonly used technique for reducing delays

associated with presenting information to a user. For example, Benedict discloses a feature in which Harvard Graphics can prepare for the display of a subsequent slide during the display of a current slide, and notify the user when the next slide is ready for display:

Signalling [sic] for the Next Slide. Depending on the power and speed of your computer, Harvard Graphics may need time to prepare the next slide in the presentation before displaying it. The Signal when next slide is ready: options in the Edit ScreenShow Effects dialog box enable you to activate an audible signal (a beep) and visual signal (a small arrow in the bottom right corner of your screen) to indicate when the next slide is ready.

(Benedict, p.441 (boldface in original).) Benedict confirms that “Harvard Graphics may need time to prepare the next slide in the presentation before displaying it” (*id.*), but by preparing the next slide during the time the user is viewing the current slide, the next slide can be displayed when needed without unnecessary delay.

64. In my opinion, as I will explained below, Benedict renders obvious claim limitations reciting that “during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed,” because during the presentation of the content resources of a current slide, the resources in at least a subsequent slide are being

accessed in order to “prepare the next slide in the presentation before displaying it.” (Benedict, p.441.) But Benedict does not disclose its internal details about how the next slide pre-loading technique works, and for this reason, I have also cited to Wolfe, which discloses a system for pre-loading information while the user is viewing other information.

65. Wolfe explains that a set of documents can be identified for the user, for example, by using the results of a search query. (Wolfe, 1:53-57 (“Whatever querying technique is used, a search engine uses the query [to] define a class of documents within the database, and locates each of the documents within the class. The user then browses the search results by ‘paging through’ one or more of the search documents.”).) Wolfe laments that “today’s database systems are plagued by the problem of slow document retrieval.” As Wolfe explains:

Consequently, today’s database systems are plagued by the problem of slow document retrieval. After a query is processed and search documents are identified, the user begins browsing the search results by studying the first view (or screen of information) from one of the search documents. The user then either “pages-down” to the next view within the same document, or moves to another search document. Each of these moves requires that new information be retrieved from the database before it can be displayed on the screen. Since the retrieval time is substantial... the database system is slow in responding to the

user's request for the information.

(Wolfe, 2:1-12.) Wolfe addresses this issue through a preloading technique in which, while the user views certain information, other information the user will likely to need at a later point is loaded:

This need can be satisfied by effectively utilizing the time the user spends studying the information on the display screen. In a database system (or document retrieval system) in one embodiment of the present invention, information that the user is likely to eventually request is preloaded into memory while the user is viewing other information. The present invention takes advantage of the fact that it is possible to accurately predict the information that the user will eventually request be shown on the display. The present invention also takes advantage of the fact that the time that the user spends viewing displayed information is often sufficient to advantageously preload a substantial amount of information.

(Wolfe, 2:23-35.) Although an exemplary embodiment in Wolfe involves an on-line legal database, Wolfe makes clear that his techniques are broadly applicable to any type of documents that can be retrieved and presented to the user. (Wolfe, 4:20-23 (“But a ‘document’ need not only be text—it could be a picture, video clip, or information in another format, such as might be available in a multimedia encyclopedia.”).) Wolfe also makes clear that his techniques are applicable to both

single-user databases as well as networked databases accessible by multiple users simultaneously. (Wolfe, 3:60-4:8.) As I will explain below, Wolfe explains that preloading is particularly appropriate in a scenario in which the system can predict what information the user will subsequently require. (Wolfe, 2:23-35.) The systems of Shimizu and Benedict clearly present such a scenario, as explained, because nodes are presented to the user in a predetermined (and thus predictable) sequence based on the link information. Wolfe thus provides a clear advantage in the sense that, during the duration between sequential nodes when the user is reviewing the content, the system can prepare for the display of the next node. I will provide more detail about Wolfe in my discussion of the claim limitations below.

2. Additional References

(a) Guinan (EX1008)

66. **Guinan** is an article entitled “Information Retrieval from Hypertext Using Dynamically Planned Guided Tours.” I am informed that Guinan qualifies as prior art to the challenged claims because it was published before the earliest priority date of the challenged claims.

67. My Declaration relies on Guinan primarily for independent claims 22 and 46, which relate to performing a search to locate search results in response to a user request. As explained in more detail below, Guinan discloses a technique for

(a) allowing a user to enter a natural language search query and then (b) searching for hypertext nodes in response to the query to create a “guided tour” for the user. (Guinan, pp.124-125.)

68. Guinan explains “[i]n using any hypertext system an end-user or browser will encounter many technical problems that have been well-documented elsewhere,” such as “disorientation and the quality of the information retrieved by the user from the hypertext.” (Guinan, p.122, §1.) This problem with hypertext was well-known and, as acknowledged in Fox, one of the key reasons the industry adopted on-line searching in the Web context. (Fox, pp.10-11 (“Hypertext makes it easy to get lost... [W]hat if you want to find pages dealing with a specific topic? Luckily, the Web is full of indexes. An index is a special Web page that allows you to type in a search term or keyword. The index then spits out a list of Web pages matching your search.”).)

69. Guinan states that, to address problems of hypertext systems:

We propose a method for dynamically generating a guided tour through a hypertext in response to a user’s query which will plan a tour through a hypertext covering all the nodes the system believes relevant to the user’s information need as expressed in a natural language query, which eases disorientation problems by using the guided tour metaphor... and which takes advantage of typing of information links in the hypertext

to present the nodes on the tour in a logical fashion. Effectively we extend the guided tour metaphor by including the information retrieval features... and we add in logical sequencing of nodes on the tour to complete the process.

(Guinan, p.123, §1 (Introduction).)

70. The first step in the proposed method is for the user to enter a “natural language” search query. (Guinan, p.124, §3.) The hypertext nodes are then searched for keywords matching the query, which are scored using a weighting algorithm. (*Id.*; see also *id.*, p.129, §6 (“*Finding information*: When the user enters a search query in the form of a natural language statement it is matched to the nodes in the hypertext using standard information retrieval techniques.”) (italics in original).) Guinan explains that, in weighting hypertext nodes, search terms that appear less frequently throughout the collection are given more weight, and as such, hypertext nodes containing more of these less frequent terms are given higher weights. (*Id.*) Hypertext nodes are also weighted based on the number of connections they have to other nodes. (*Id.*) Hypertext nodes must satisfy a threshold value to be included in the tour. (Guinan, p.125 (“All nodes whose weight is greater than or equal to this cut-off or threshold value would be selected for inclusion in the tour.”).)

71. Once matching hypertext nodes have been identified, a guided tour is

assembled. “At this stage, the nodes for the guided tour have been dynamically chosen in response to a user’s query and some form of logical ordering has to be imposed on the selected tour nodes.” (Guinan, p.125, §3.) Guinan discloses an ordering procedure that takes into account linking relationships between the matching nodes. (*Id.*) “This procedure is followed until every node selected for inclusion in the tour has been positioned correctly relative to each other node in the tour.” (*Id.*)

72. Guinan identities several advantages of her guided tour technique. For example, “[u]sers don’t need to worry about getting lost or confused in the hypertext as all they need to do is follow the tour which is generated in response to their query.” (Guinan, p.129, §6.) I will provide more information about Guinan in my discussion of claim 22 below.

(b) Miller (EX1010)

73. **Miller** is a book entitled *Using CompuServe, Second Edition*. I am informed that Miller qualifies as prior art because it was published before the earliest priority date of the challenged claims.

74. My Declaration cites Miller in combination with Guinan primarily for claim 46, which recites “providing a predefined list of categories,” and “providing a plurality of search results based on user selections and their corresponding URLs in

response to a user request related to a category among said predefined list of categories.” Miller discloses the claimed category selection and performing a search to obtain results based on selected categories.

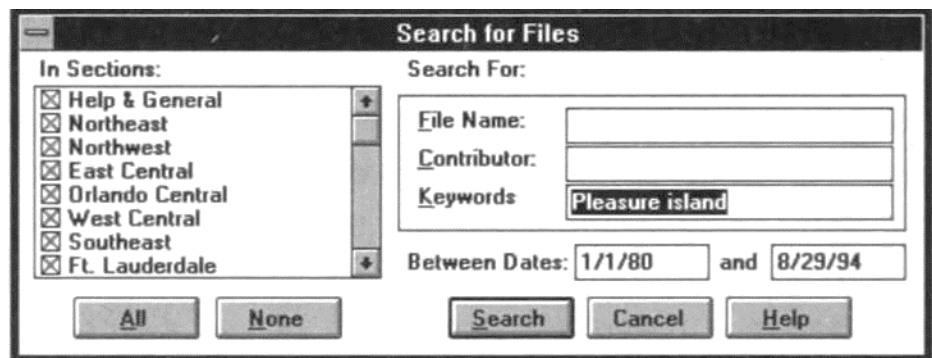
75. As its title implies, Miller explains how to use CompuServe, an early commercial on-line service that was popular in the mid-1990s. “CompuServe is like a giant electronic community.” (Miller, p.1.) CompuServe provided a feature known as “forums,” which were on-line communities where members could exchange information and messages about particular topics of interest. (Miller, p.29 (“You can find forums on just about any topic, from scuba diving to C programming.”); *id.*, p.30 (“A forum is like one of those after-school clubs you belonged to back in high school. A bunch of people with the same interests get together to talk and exchange information.”).) Forums included a message section, “where forum members exchange messages with each other,” and a file library, “where forum files are stored.” (Miller, p.34.)

76. Miller discloses that, “[i]n addition, every forum’s message sections and file libraries are comprised of multiple *sections*. Each section represents a more focused area of interest within the forum topic.” (Miller, p.35 (*italics in original*).) “For example, the Florida Forum has sections for Ft. Lauderdale, Tampa, St. Petersburg, and other specific locations.” (*Id.*) “The MacMillan Computer

Publishing Forum (where you talk to Que [the publisher of Miller] has sections for Word Processing, Databases, Programming, and several other specific topics.” (*Id.*)

77. Miller also discloses features of CompuServe that allow users to search forums and sections within the forums. Figure 3-9 shows a user interface dialog box that allows a user to perform such a search:

Fig. 3.9
Use the Search for
Files dialog box to
search for specific
forum files.



(Miller, p.42, Fig. 3-9; *see also id.*, p.42 (“A better way to find files... is to click the Search Libraries button in the toolbox. When you do this, a Search for Files dialog box appears (see fig. 3.9).”).) The dialog box, as shown, allows the user to select one or more sections to search, and enter additional information such as keywords. (*Id.*) “When you click the Search button, the forum’s libraries are searched for any and all files that match your search criteria.” (Miller, p.43 (boldface in original); *see also id.*, p.419 (showing similar example).)

78. I will provide more details about Miller in my discussion of the claim limitations below.

(c) **Filepp (EX1007)**

79. **Filepp**, U.S. Patent No. 5,347,632, is entitled “Reception System for an Interactive Computer Network and Method of Operation.” I am informed that Filepp qualifies as prior art because it was filed on July 28, 1989 and issued on September 13, 1994, before the earliest priority date of the challenged claims.

80. My Declaration cites Filepp primarily for limitations of dependent claims 30-31 relating to the delivery of “**an advertising message**” (claim 30) and dynamic content “**in accordance with a user profile**” (claim 31). Filepp discloses an “interactive computer system that enables a user to display desired information, such as news, financial and cultural information, and perform desired transactional services, such as banking and shopping, through any of a plurality of types of personal computers.” (Filepp, Abstract.) Filepp was originally assigned to Prodigy Services Company, an early provider of interactive on-line services that competed with companies such as America Online (AOL) before widespread adoption of the Web.

81. The interactive system in Filepp displays “pages” to the user, with each page divided into a series of partitions as shown in Figure 3a:

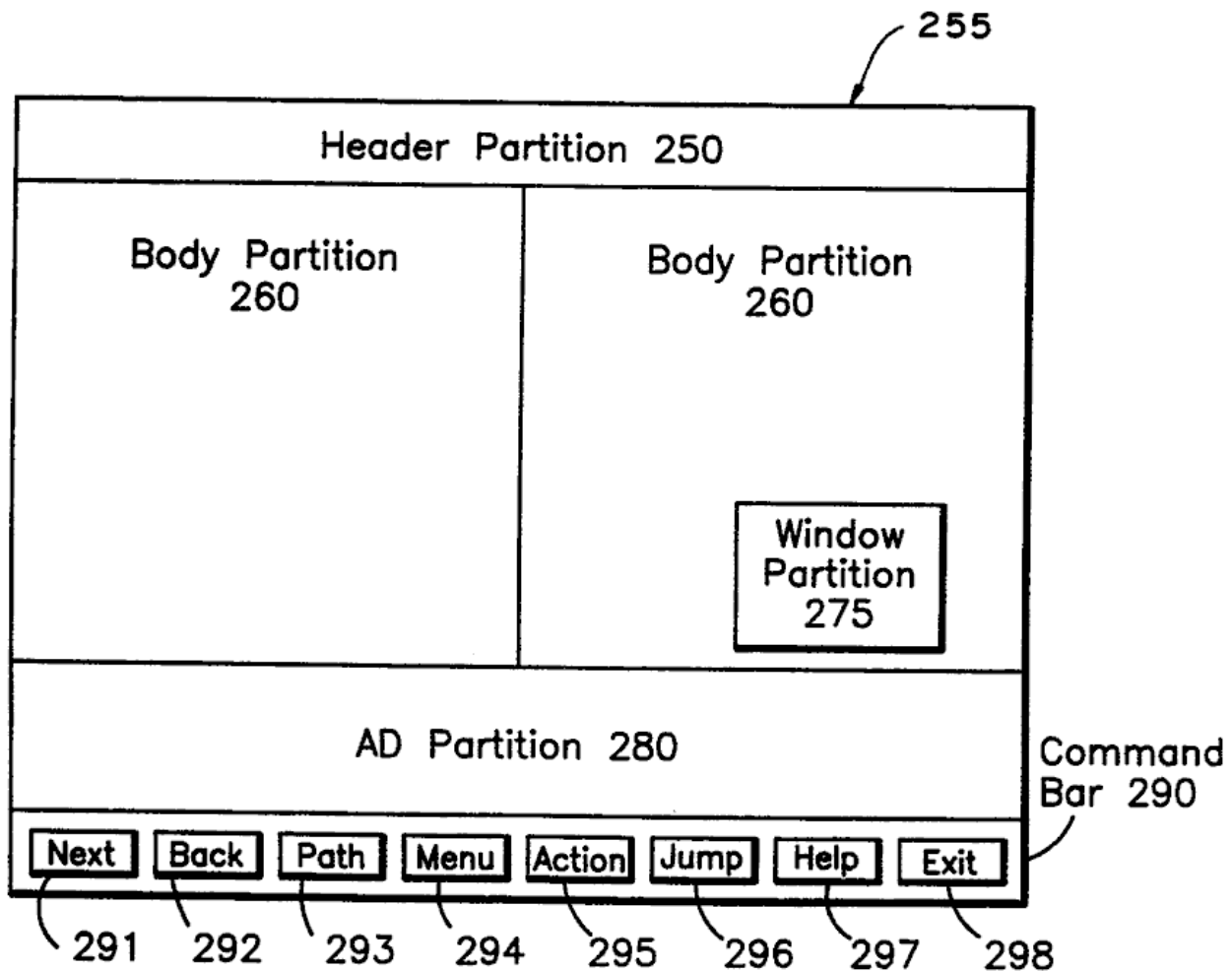


FIG. 3a

(Filepp, Fig. 3a.) “With reference to FIG. 3a, each page **255** is formatted into page partitions **250**, **260**, **280**, and **290** (not to be confused with applications partitions).”

(Filepp, 8:68-9:3.) Header partition **250** “typically conveys information on the page’s topic or sponsor,” and body partitions **260** and window partition **275** “gives the information and transactional content of the page.” (Filepp, 9:10-16.) AD

partition **280**, as its name implies, displays advertisements provided over the network. (Filepp, 9:27-30, 9:30-34 (“Advertisements **280** may be presented to the user on an individual basis from queues of advertisements that are constructed off-line by business system **130**, and sent to file server **205** where they are accessible to each RS **400**.”), 12:64-66.)

82. With respect to the particular advertisements that can be displayed to the user, they may be selected based on user profile information such as information about the user’s past activities or its demographic information, ZIP code, etc.:

Individual queues of advertisements are constructed based upon data collected on the partitioned applications that were accessed by a user, and upon events the user generated in response to applications. The data are collected and reported by RS **400** to a data collection co-application in file server **205** for later transmission to business system **130**. In addition to application access and use characteristics, a variety of other parameters, such as user demographics or postal ZIP code, may be used as targeting criteria. From such data, queues of advertisements are constructed and targeted to either individual users or to sets of users who fall into certain groups according [sic; to] such parameters.

(Filepp, 9:35-47.)

83. I will provide more information about Filepp in my discussion of the claim limitations below.

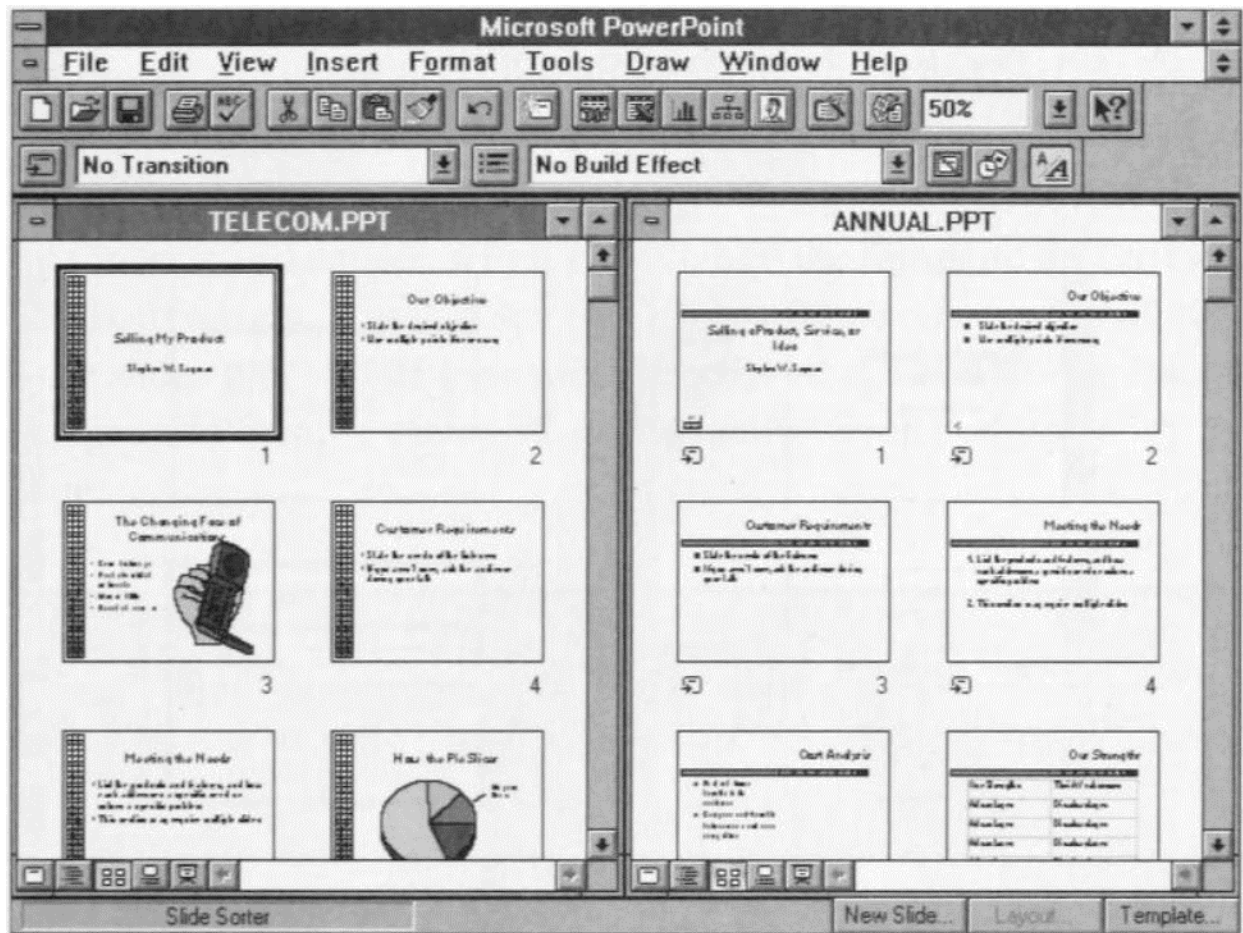
(d) Sagman (EX1006)

84. Sagman, is a book entitled *Running Microsoft PowerPoint 4 for Windows*. I am informed that Sagman qualifies as prior art because it was published before the earliest priority date of the challenged claims.

85. Sagman describes a software program known as Microsoft PowerPoint that provided functionality similar, in some respects, to the Harvard Graphics product described above. PowerPoint allowed users to create a presentation including a series of slides. (Sagman, pp.6-7.) My Declaration relies on Sagman primarily for independent claim 63, which recites the use of drag-and-drop techniques in connection with the creation of a show structure.

86. Sagman discloses a “Slide Sorter” feature that is similar to the Harvard Graphics “Slide Sorter” feature described by Benedict above. But the Slide Sorter feature in Sagman has an additional feature, *i.e.*, the ability to show two side-by-side presentations on the screen and to copy slides from one presentation to another using drag-and-drop:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2



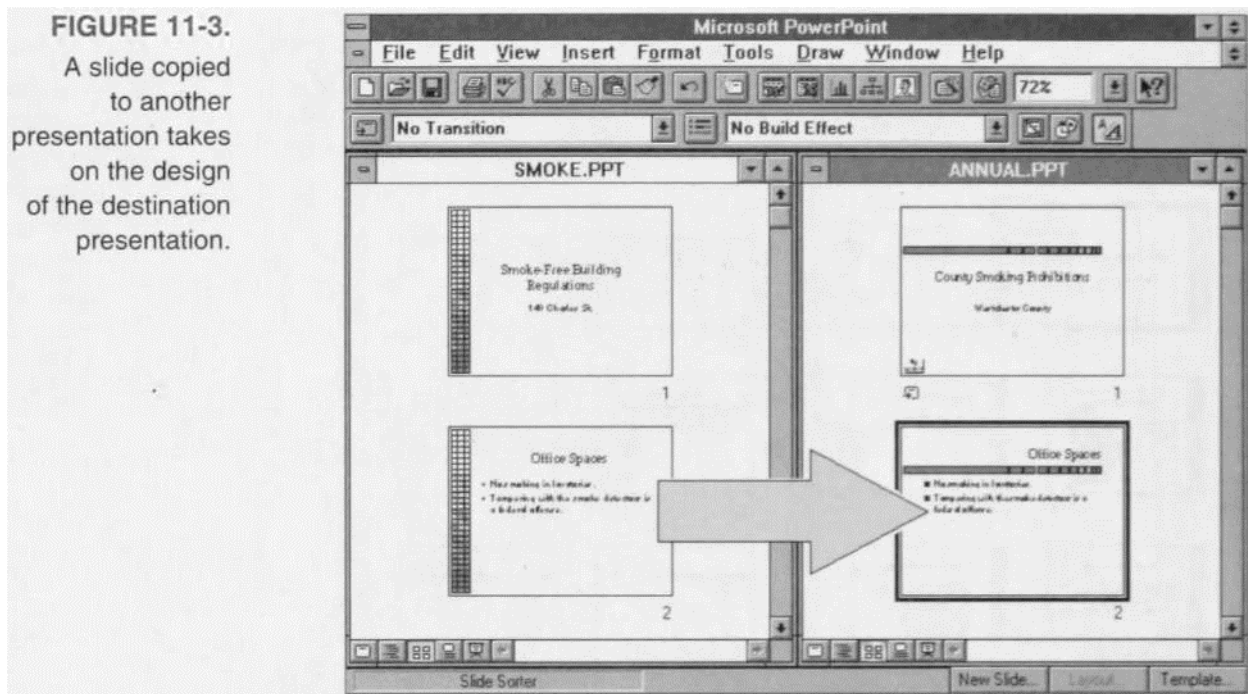
(Sagman, p.311.) The screen display above shows two PowerPoint presentations (TELECOM.PPT and ANNUAL.PPT) displayed side-by-side in two windows, with icons representing the individual slides in each presentation. Sagman explains that, using standard drag-and-drop techniques, the user can drag a slide from one presentation into the other:

After the presentations are arranged side by side, you can move a slide from one presentation to another by clicking on a slide in one presentation and dragging it to the other presentation. The moved slide

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

takes on the design of the destination presentation's template. To copy a slide rather than move it, hold down the Ctrl key as you drag the slide to the destination presentation. In Figure 11-3 on the next page, the bottom slide has been copied from one presentation to another. Note its appearance in each presentation.

(Sagman, p.313.) Figure 11-3, referenced above, shows the result of copying a slide from one presentation into another:



(Sagman, Fig. 11-3, p.312.) In this example, the slide “Office Spaces” was copied from SMOKE.PPT into the ANNUAL.PPT presentation. As shown, the slide in the target presentation retains the same content as the original slide but takes on the design of the target presentation. I will provide additional detail about Sagman in

my discussion of the claims below.

(e) Robertson (EX1009)

87. **Robertson**, U.S. Patent No. 6,486,895 B1, is entitled “Display System for Displaying Lists of Linked Documents.” I am informed that Robertson qualifies as prior art because it was filed on September 8, 1995, before the earliest priority date of the challenged claims.

88. My Declaration cites Robertson for claim 65 reciting “**presentation of content from at least two nodes presented simultaneously.**”

89. Robertson discloses “[a] display system for displaying lists of linked documents,” in which “[a]n example of linked documents are hyper-text documents in a hyper-text system,” such as the Apple HyperCard system. (Robertson, 3:24-27.) Robertson explains that his currently preferred embodiment uses “lists of linked documents obtained from the portion of the Internet known as the World Wide Web (hereafter the Web).” (Robertson, 2:33-36.)

90. Robertson discloses a technique for displaying a web-based book that presents the web pages in sequence. (Robertson, Abstract, 2:14-25.) “The list of pages is displayed in a book metaphor, termed WebBook, on a computer controlled display system... When displayed, two active pages are displayed at one time.” (Robertson, 2:24-28.) Figure 3 of Robertson shows an exemplary WebBook view

showing two web pages visible in a side-by-side fashion:

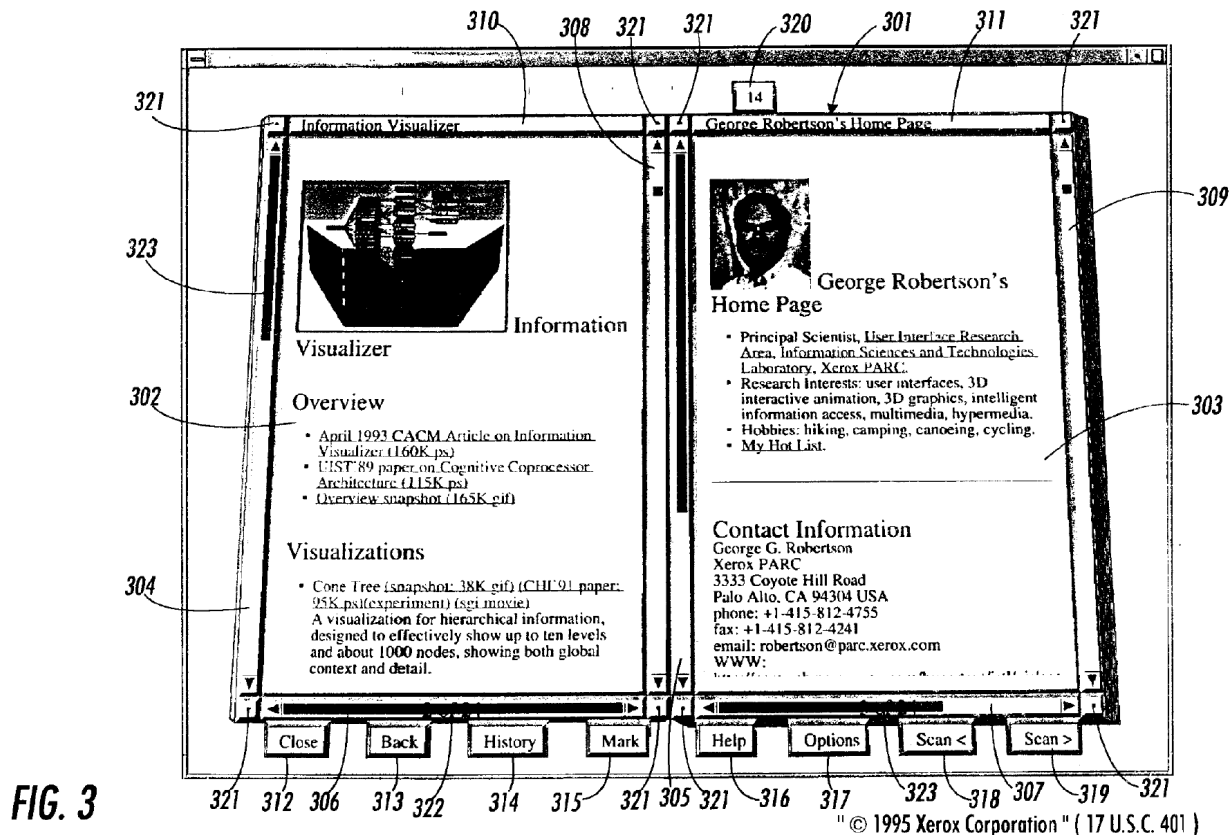


FIG. 3
(Robertson, Fig. 3.) As explained in Robertson:

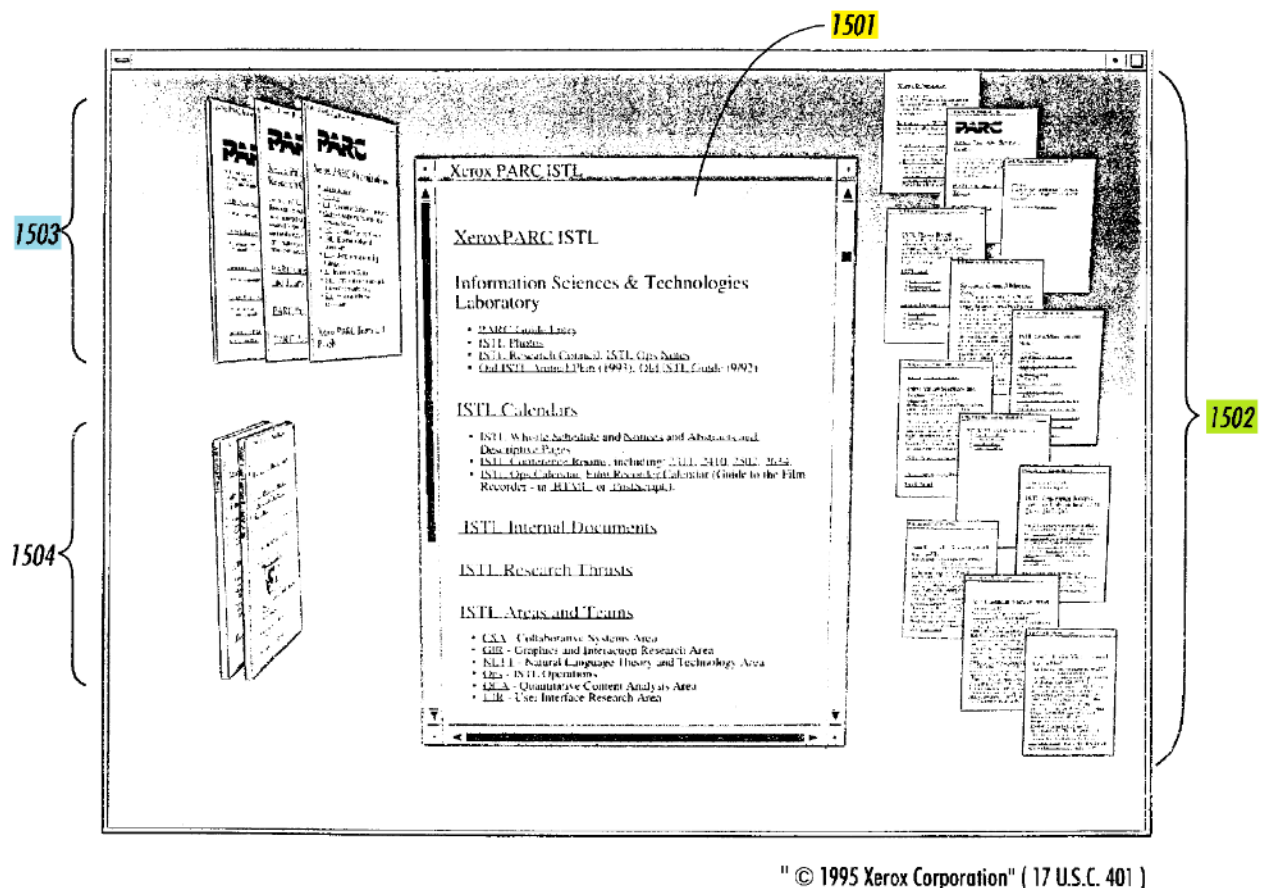
FIG. 3 is an illustration of a WebBook as displayed on a display. Referring to FIG. 3, the WebBook 301 is comprised of a first page 302 and a second page 303, each having an identical structure. The first page 302 is an instance of a left page and the second page 303 is an instance of a right page.

(Robertson, 6:15-19; *see also id.*, 6:28-30 (“The content and layout of the page is determined by the HTML macros defining the page.”).) Robertson explains that, in order to display web pages in a WebBook, each Web page is converted into a page

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

object. (Robertson, 7:65-66.) "Each page is provided in the order in which it is to appear in the WebBook. Accordingly, each page object will represent either a left page or a right page in the WebBook." (Robertson, 7:66-8:2.)

91. Robertson is also not limited to simply viewing two pages at the same time. Another embodiment shown in Figure 15 shows the ability to view "thumbnails" of potentially all of the pages in a WebBook:



(Robertson, Fig. 15 (highlighting added).)

92. Figure 15 above shows a WebBook providing “a feature of a complete Web Browsing system.” (Robertson, 10:15-16.) Area 1501 in the center shows the current web page viewed by the user. (Robertson, 10:17-19.) Area 1503 to the left “contain[s] pages which have been viewed by the user.” (Robertson, 10:22-24.) Area 1502 on the right contains “a collection of ‘thumbnail’ images of pages,” which have “not yet been viewed by the user.” (Robertson, 10:19-22.) “As the user views pages and skips to different pages, the prior current page is moved to the area 1503.” (Robertson, 10:24-26.) “It should be noted that the collection of pages, i.e. the current page 1501 and the pages in the areas 1502 and 1503 themselves may be pages from a WebBook.” (Robertson, 10:26-28.)

93. I will provide more information about Robertson in my discussion of the claim limitations below.

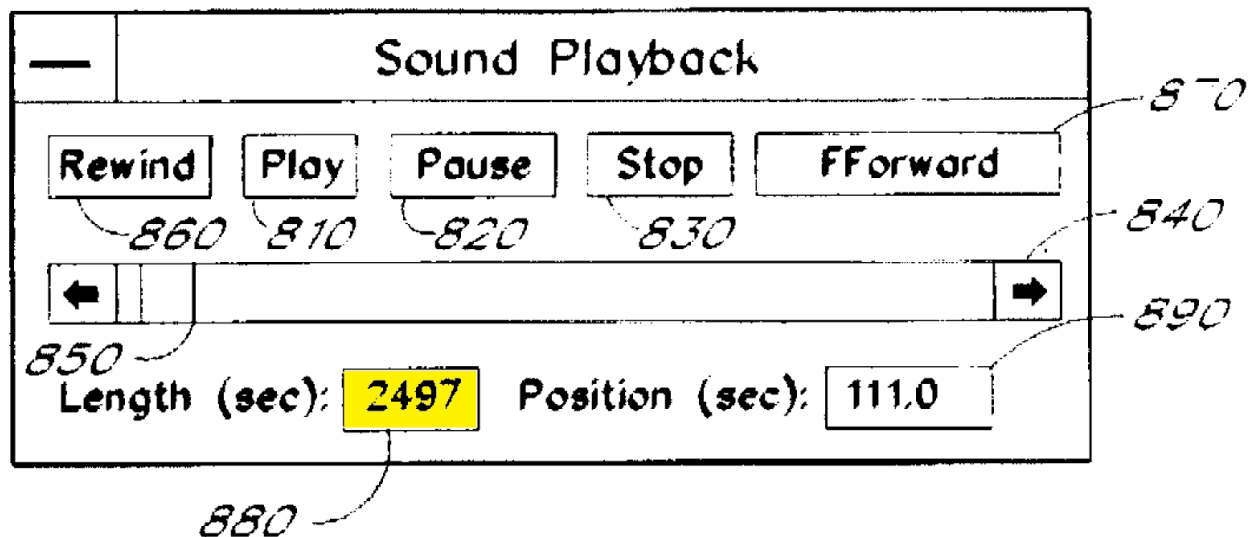
(f) Glaser (EX1019)

94. **Glaser**, U.S. Patent No. 5,793,980, is entitled “Audio-On-Demand Communication System.” I am informed that Glaser qualifies as prior art because it was filed on November 30, 1994, before the earliest priority date of the challenged claims. My Declaration cites Glaser for dependent claims 11-16.

95. As explained above, Benedict discloses an Edit ScreenShow Effects user interface that displays a text box for the user to set or change the display time

for one or more slides. (Benedict, p.439, Fig. 17.1; *see also id.*, p.440 (“You can set a display time... by typing a time in the Slide display time: text box.”).) Benedict therefore discloses that “**interactively variable duration information is displayed to the user**” because the display time would have been displayed to the user, for example, in a text box when the user enters or edits the display time. But Benedict does not disclose that the interactively variable duration information is displayed to the user “**during presentation of said content.**” I have therefore cited Glaser in combination with the prior art for Ground 1 for both claims 11 and 12.

96. Glaser discloses an audio-on-demand system for transmitting audio tracks to a user computer for playback. (Glaser, Fig. 1, 1:54-57, 4:63-5:12.) My Declaration cites Glaser for a narrow purpose – to disclose certain user interface features relating to the playback of audio content as shown in Figure 8A:



(Glaser, Fig. 8A (partial figure; highlighting added).) Figure 8A shows a user interface with control buttons for familiar playback functions including Play **810**, rewind **860**, Pause **820**, Stop **830**, and so forth. Scroll box **850** allows the user to seek to a particular location within the audio clip and begin playing the clip from that location. (Glaser, 10:22-30.) Field **890** indicates the amount of time that has elapsed for the audio clip currently being played. (Glaser, 18:19-30.) For purposes of my Declaration, the most relevant feature in Figure 8A is length field **880**, which displays the length of the audio clip. (Glaser, 19:19-26.)

97. I have also cited Glaser for the control button limitations in claims 13-16. I will provide more detail about Glaser in my discussion below.

B. Ground 1: Claims 1-11, 14, and 16-21 Are Obvious Over Shimizu, Fox, Benedict and Wolfe

1. Independent Claim 1

98. For reference, claim 1 recites:

1. A method for providing content via a communications network on a user's computer, obtained from a plurality of resources in an organized arrangement comprising the steps of:
 - [a] creating a show structure of nodes, each node identifying a resource from a plurality of accessible resources, at least some of which are network accessible resources;

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

- [b] without requiring user input, automatically accessing a plurality of said network accessible resources identified by a corresponding node; and
- [c] causing content from said network accessible resources corresponding to each of said accessed resources to be automatically accessed and presented through said user's computer in accordance with said show structure such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed,
- [d] wherein said step of creating further comprises the step of providing interactively variable duration information, representing the duration that said content corresponding to said resource is presented so as to enable a user to vary said duration.

('932, 17:42-64 (Claim 1; bracketed notation (e.g., [a]) added).)

99. The preamble recites “[a] **method for providing content via a communications network on a user’s computer, obtained from a plurality of resources in an organized arrangement....**” It is unclear if the preamble of claim 1 recites a claim limitation, but even if it does, it is disclosed by the prior art.

100. Shimizu discloses a method of providing content on a user’s computer for display, content such as text, pictures, motion pictures, etc.:

When the entity information (character string, image, sounds, motion pictures) is acquired from the node information entity management part

8, the content of the acquired information is reproduced and displayed in the node content display area 81 of the window 80 by applying the program which is associated with the pointer in the display procedure slot 62 of the node information structure 9.

(Shimizu, 8:52-59.) Figure 8 shows a user interface for presenting node content:

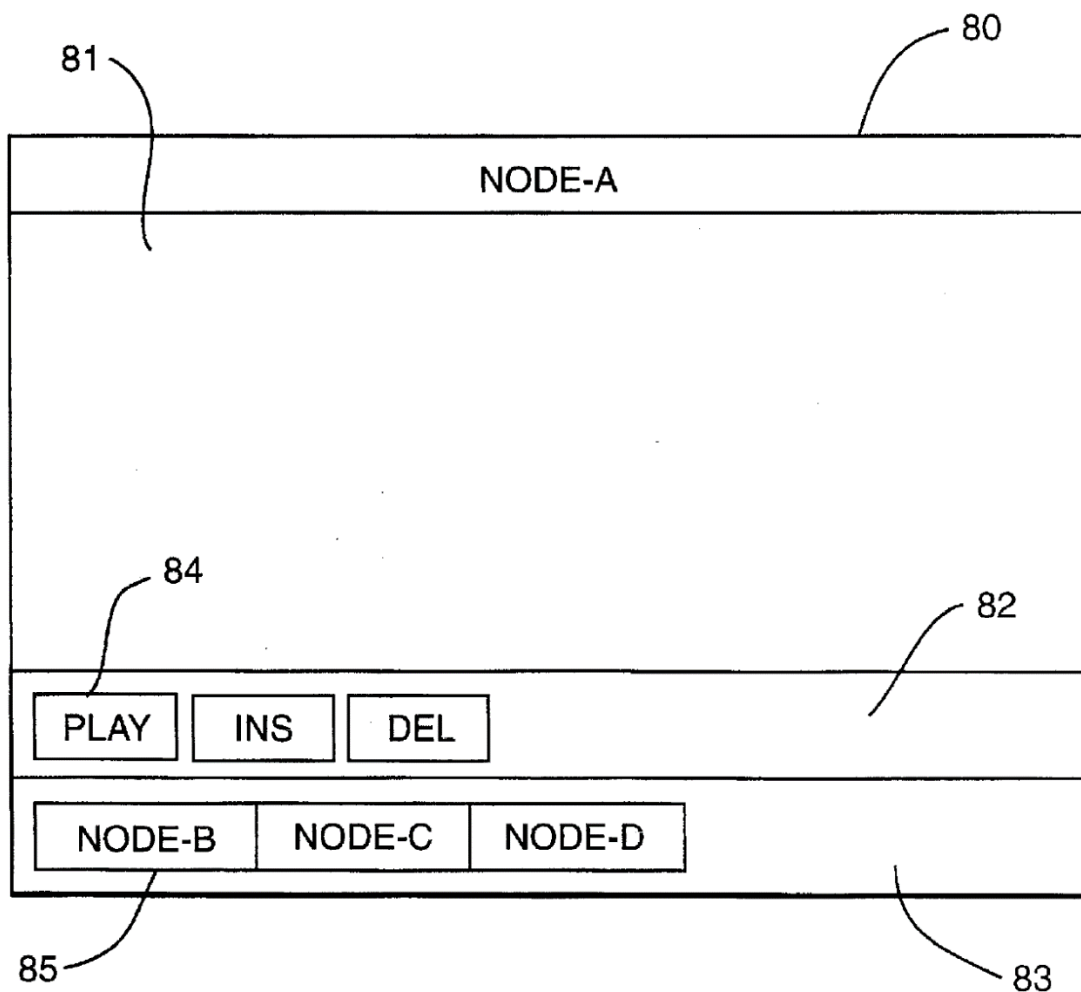


FIG. 8

(Shimizu, Fig. 8, 7:12-14 (“FIG. 8 is a view of a typical user interface screen display provided by an application program running on the embodiment.”).) As noted, when

the user presses PLAY in Figure 8 (one of buttons **84**), the content associated with the current node (in this case NODE-A) is provided to and displayed in area **81** of window **80**. (Shimizu, 8:11-25, 8:49-59.) Shimizu therefore discloses “[a] **method for providing content... on a user’s computer,**” as recited.

101. The content in Shimizu is also “**obtained from a plurality of resources,**” as recited in the preamble. This is because the character strings, images, motion pictures, etc., in Shimizu are obtained from stored information (for example, in computer files), from which the content can be derived. It would have been obvious that the content in Shimizu is stored in files, as these are basic units of data storage in a computer system. (See *Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.164 (“**file** A complete, named collection of information, such as a program, a set of data used by a program, or a user-created document. A file is the basic unit of storage that enables a computer to distinguish one set of information from another.”).) As will be described in more detail for claim 1 [a] below, Shimizu further discloses that, in order to display content associated with a particular node, the system uses an entity information pointer slot **63** for the node, which points to the particular content that will be retrieved and displayed to the user. (Shimizu, 6:55-57 (“The entity information pointer slot **63** retains the pointer pointing to the entity information to be displayed by the node in question.”).) As explained in

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

Shimizu:

Step **107** accesses and reads the entity information linked by the entity information pointer of the node information structure designated by the pointer pointing to the acquired node. When the entity information (character string, image, sounds, motion pictures) is acquired from the node information entity management part **8**, the content of the acquired information is reproduced and displayed in the node content display area **81** of the window **80** by applying the program which is associated with the pointer in the display procedure slot **62** of the node information structure **9**.

(Shimizu, 8:49-59.) Shimizu thus discloses that the displayed content is “**obtained from a plurality of resources,**” with the resources corresponding to the sources of content identified by the entity information pointer **63**.

102. As will be explained in more detail below, under the proposed combination with **Fox**, it would have been obvious that the entity information pointer **63** could take the form of a Uniform Resource Locator (URL) pointing to a resource (web page, image, etc.) on the Web. (Fox, p.13 (“A *URL* (or *Universal Resource Locator*) is the address of a document, file, or other Internet resource... The World Wide Web gives every article, newsgroup, database, computer, file, or other Internet fragment its very own URL.”); *see also id.*, pp.326-327 (describing IMG tag identifying location of image resource).) As explained for claim 1[a]

below, it would have been obvious under the proposed combination that the content would be obtained “**via a communications network**,” *i.e.*, the Internet.

103. Finally, Shimizu discloses providing content on a user’s computer “**in an organized arrangement**,” as claimed. This is because the content of each node is presented in a particular order or sequence based on the sequencing of nodes:

The processing above is carried out so that the node C is displayed followed by node B and node D, in accordance with the order of the link information.

(Shimizu, 9:1-4.)

As described, the hypertext apparatus of the invention offers a user interface capable of putting into a desired order the links between the nodes to be managed and of manipulating the established link order. This makes it easy to create illustratively an application program allowing the display of node contents to be switched in a desired order and at predetermined intervals on the hypertext apparatus.

(Shimizu, 9:60-66; *see also id.*, 5:10-14.) Shimizu thus discloses providing and displaying content “**in an organized arrangement**” because, as noted above, node content is displayed to a user in a particular sequence as indicated by the link information.

- (a) **“creating a show structure of nodes, each node identifying a resource from a plurality of accessible resources, at least some of which are network accessible resources;” (Claim 1[a])**

104. The proposed combination renders obvious this claim limitation. The **“show structure of nodes”** in Shimizu corresponds to a set of nodes and their links or connections to other nodes, as explained in more detail below. Shimizu explains that node information management part **5** and link information management part **6** are used to create the show structure of nodes:

The node information management part **5** manages a plurality of node information structures holding the information about each node. The link information management part **6** manages a plurality of link information structures defining the relations between the nodes. The link order manipulation part **7** changes the order of a plurality of link information structures **10** referenced by the node information management part **5** under the directions of the program execution part **3**.

(Shimizu, 6:30-39.) A conceptual example of a show structure of nodes in Shimizu is shown in Figure 9:

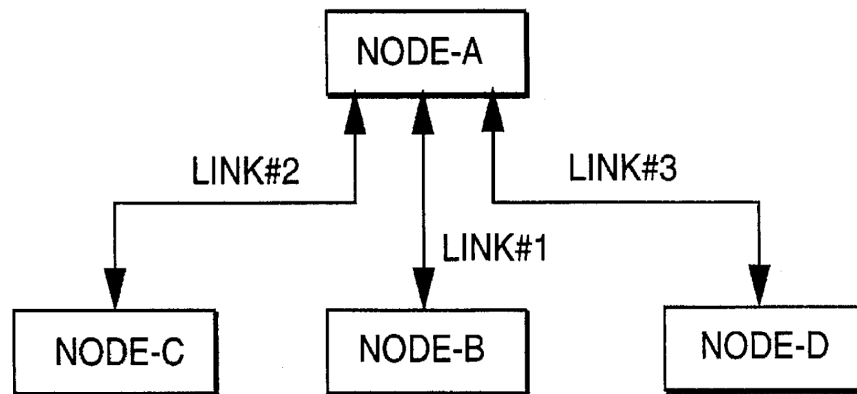


FIG. 9

(Shimizu, Fig. 9, 7:49-51 (“FIG. 9 is a conceptual view of typical nodes and links and the relations in effect therebetween when the order of link information is changed.”).) In this example, NODE-B, NODE-C, and NODE-D are linked to NODE-A by LINK#1, LINK#2, and LINK#3, respectively. (Shimizu, 7:51-54.) In this example, therefore, NODE-A would be displayed first followed by NODE-C, NODE-B, and NODE-D. (Shimizu, 7:54-65.) Figure 10 shows a more concrete representation of the show structure from Figure 9:

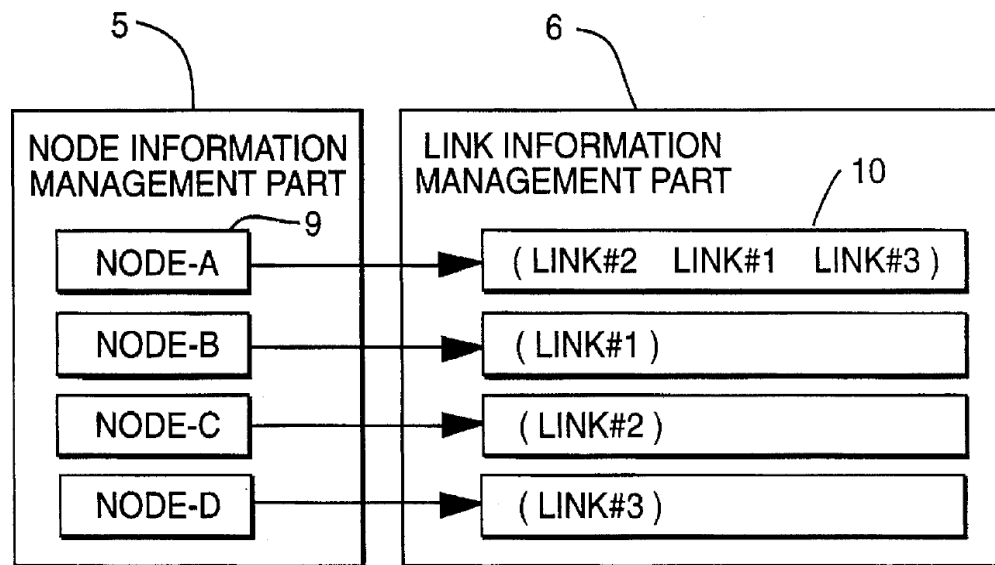


FIG. 10

(Shimizu, Fig. 10, 7:67-8:5 (“FIG. 10 is a schematic view of a typical data constitution in which the hypertext data of the nodes and links in FIG. 9 is stored in the node information management part and link information management part of the embodiment. As explained above, the node A is linked to the node C, node B and node D, in that order.”).)

105. As explained for the preamble, the node and link information is stored, respectively, in node and link information structures. (Shimizu, Fig. 6 (node information structure), Fig. 7 (link information structure).) Shimizu thus discloses creating “**a show structure of nodes**,” which corresponds in Shimizu to an arrangement or set of nodes that define their associated content and their links to

other nodes. With respect to the act of “**creating**” the claimed show structure, Shimizu discloses a number of techniques for adding new nodes and rearranging the order of existing nodes. (Shimizu, *e.g.*, 7:20-48, 9:5-59.) It would have been obvious to an ordinarily skilled artisan that the show structure in Shimizu was “**creat[ed]**,” as it must have been brought into existence in order to perform the node display and other functions described in Shimizu.

106. The proposed combination also discloses “**each node identifying a resource from a plurality of accessible resources.**” As noted, the node information management part **6** in Shimizu system holds information about individual nodes. Figure 6 shows an example of a node information structure **9**:

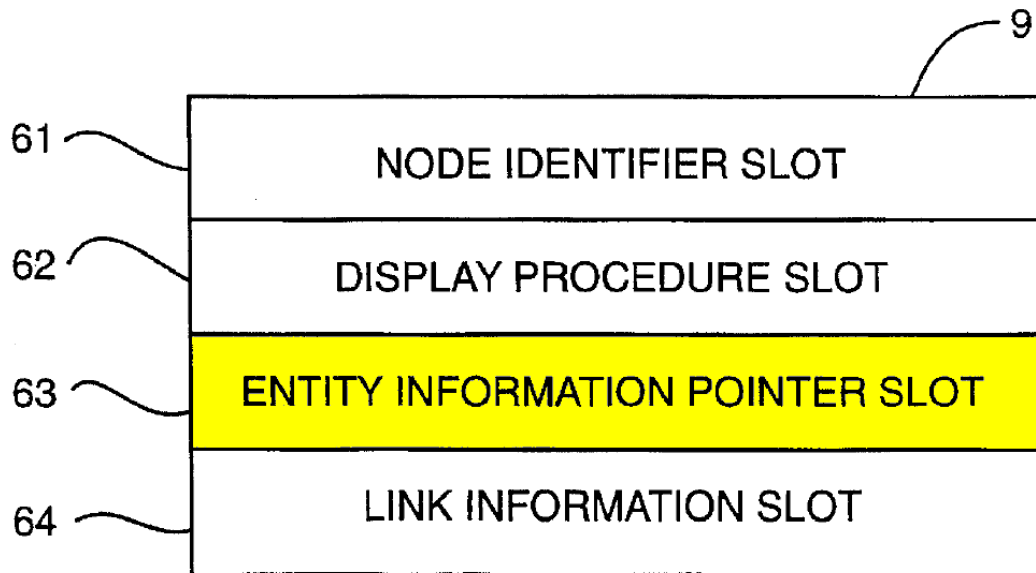


FIG. 6

(Shimizu, Fig. 6 (highlighting added), 6:44-45 (“FIG. 6 is a view showing members of a node information structure held by the node information management part 5.”).)

Entity information pointer slot **63** shown in yellow holds an identification of the resource associated with the node:

The entity information pointer slot **63** retains the pointer pointing to the entity information to be displayed by the node in question. The entity information thus pointed to may be a character string, image information, motion pictures or sounds.

(Shimizu, 5:55-59.) As noted previously, the system in Shimizu uses the entity information pointer to identify and retrieve the content of a node that will be displayed. (Shimizu, 8:49-59.) The nodes in Shimizu thus through its node

information identifies a particular content resource (e.g., entity information such as text, image, motion picture, sound, etc.) among a number of available resources.

107. It would have also been obvious based on Fox that “**at least some of which are network accessible resources,**” as recited at the end of claim 1[a]. Shimizu discloses that resources could have been stored on “primary storage” or “secondary storage” (e.g., Shimizu, 6:39-43, 2:49-53). The term “secondary storage” was commonly understood as referring to any storage other than the computer’s “primary storage,” i.e. its volatile storage such as the random access memory (RAM) directly accessible to the computer’s processor. (See *Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.351 (“**secondary storage** Any data storage medium other than a computer’s random access memory—typically, tape or disk. *Compare* primary storage”) (bold and italics in original), see also *id.*, p.314 (“**primary storage** Random access memory (RAM); the main general-purpose storage region to which the microprocessor has direct access. A computer’s other storage options, such as disks and tape, are called secondary storage or (sometimes) backing storage.”).) Accordingly, an ordinarily skilled artisan would have understood “secondary storage” includes a storage medium accessible through a network, and as such, it would have been obvious to an ordinarily skilled artisan based on Shimizu alone that content resources could have been accessible via

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

a communications network. But Shimizu does not expressly disclose an example in which resources are expressly described as network accessible, so to remove any doubt about this issue, network accessible resources would have been obvious in further view of **Fox**.

108. Fox is a textbook describing various techniques for creating web pages. Fox confirms that a web page “can be anything you imagine it to be: a hypertext document, graphics sound, animation, tabular data, software files, a discussion group, a search index, a specialized program, and more.” (Fox, p.8; *see also id.*, p.9 (“The World Wide Web uses hypertext to link all its pages together.”).) As was well-known to ordinarily skilled artisans, the Web provided an example of a hypertext system. (Fox, EX1005, pp.8-10 (describing the Web as an example of a hypertext system).) As was basic knowledge to ordinarily skilled artisans, and even laypersons, each resource on the Web was identified by a Uniform Resource Locator (URL). (Fox, p.13 (“A *URL* (or *Universal Resource Locator*) is the address of a document, file, or other Internet resource... The World Wide Web gives every article, newsgroup, database, computer, file, or other Internet fragment its very own URL.”); *see also id.*, pp.326-327 (describing IMG tag identifying location of image resource).)

109. In fact, Fox makes clear that URLs could be used to identify resources

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

to carry out the same functions as Shimizu, *i.e.*, to create sequential presentations such as the ordered hypertext display in Shimizu or the ordered show presentations in Benedict. For example, Fox discloses a technique in which a web page, after a specified amount of time, can cause the web browser to “refresh” and load another web page, which can be used to create slide show-type presentations of web content:

To load up a different document after a set amount of seconds, simply specify a full URL using the URL attribute. For example, to load the page at <http://www.smartpants.edu/sample.html>, after a minute, use the following markup as the first line of your HTML document:

```
<META HTTP-EQUIV="Refresh" CONTENT=60  
URL= http://www.smartpants.edu/sample.html>
```

You can use this trick to flip between two documents, animate a sequence of images, or create an automatic slide show of many Web pages.

(Fox, p.316 (italics in original; underlining added).) The concept of “refreshing” a web page was well-known (and remains well known) to those familiar with web browsers, and this was (and remains) a common feature of many websites. Refreshing was commonly used to load an updated version of the *currently displayed* web page, but as shown above, the <META> tag used for this feature can specify the URL of any web page, including a different web page. In the example from Fox above, the <META> tag tells the web browser to “refresh” after 60 seconds by

switching to the web page at <http://www.smartpants.edu/sample.html>. In any case, Fox confirms that it was known to use URLs to identify resources for a presentation system such as Shimizu combination in which the resources associated with a series of nodes are presented in a particular sequence and for a specified duration.

110. ***Rationale and Motivation to Combine (Shimizu with Fox):*** It would have been obvious to combine Shimizu with Fox, predictably resulting in the Shimizu system in which each node was associated with a resource identified by a URL, such a web page, image, or other Web resource. As a result, at least some of the resources “**are network accessible resources,**” *i.e.*, resources accessible over the Internet via a URL. To the extent each of the “**plurality of resources**” is further construed to require that each resource be located at a distinct website identified by a unique domain, this would also have been obvious as this is a built-in and well-known capability of URLs.³ (Fox, pp.15-16 (listing several examples of URLs for

³ I understand Patent Owner does not contend that the claims require that each of the “**plurality of resources**” refer to a resource at different website/domain, but in the event the claim were construed in this manner, I have explained in the text why this would have been obvious. It would have been obvious that the URLs that identify the plurality of resources could each have pointed to a different website/domain. (Fox, pp.15-16 (showing exemplary HTML URLs pointing to moolah.com, testlabs.edu, strange.org, and abc.123.com).)

webpages at different website domains).) One of the components of a URL is the name of the server where the resource is located, typically expressed as a domain name (*e.g.*, “uspto.gov”). The use of URLs to identify resources on the Web would have been a straightforward adaptation; for example, entity information pointer slot **63** discussed above could have been readily adapted to hold one or more Uniform Resource Locators (URLs) identifying resources (web pages, images, etc.) on the Web. (Fox, p.13 (“The World Wide Web gives every article, newsgroup, database, computer, file or other Internet fragment its own URL.”), pp.326-327 (describing how to embed graphic files on a web page using IMG tags identifying the URL of images for display on the page).) This would have been a trivial adaptation, as the very purpose of a URL is to provide a pointer to a content resource to facilitate access by a web browser. For example, Shimizu explains that the node content information may be stored “in primary or secondary storage.” (Shimizu, 6:39-43.) A URL would have been a natural way to implement entity information pointer slot **63** because a URL provides a standardized and uniform way to identify the location of a file stored on the Internet or in local storage. (Fox, pp.15-17.)⁴

⁴ For example, with respect to local storage, a standard URL format (*i.e.*, file://) can be used to identify the location of a file on the user computer’s local storage.

111. Shimizu is an analogous reference in the same field of techniques for providing organized or sequenced presentation of content, and would have been reasonably pertinent to a number of problems facing the inventor. For example, Shimizu would have been reasonably pertinent to the problem of how to structure and organize information about nodes, in order to keep track of their associated content and the nodes to which they are linked. Similarly, Fox is an analogous reference because it falls within the field of techniques for organized or sequenced presentation of content and would have been reasonably pertinent to a number of problems facing the inventor. For example, Fox would have been reasonably pertinent to the problem of how to structure and present a series of Web resources in a particular order, and as noted, Fox also contains pertinent teachings for specifying a duration for the display using industry standard <META> HTML elements. Fox would also have been relevant to understanding the composition and function of URLs, which would have been reasonably pertinent to problems relating

(Fox, p.17 (“The file service is almost identical to the ftp service, but is a quicker way of retrieving local files. You can use file URLs to load and display sound, image, or text files....”).) URLs can also be used to identify a file on a remote server on the Internet, for example, by identifying its hostname in the URL. (Fox, p.14 (“The second part of most URLs is the full Internet address of the machine where the file is located.”).)

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

to how to identify the location on the network of content resources associated with nodes.

112. An ordinarily skilled artisan would have been motivated to combine with Fox to take advantage of the growing popularity of the Web. The Background of the challenged patent, in fact, concedes that the web was “popularized in 1993.” (’932, 1:28-31.) The challenged patent further notes that companies such as Cisco and Boeing were already conducting “web-based sales in excess of one billion dollars,” with Dell having “sales in excess of \$1 million per day via their website.” (’932, 2:45-50.) As Fox confirms, ordinarily skilled artisans had recognized the enormous commercial potential of the Web and its ability to reach millions of potential users. (Fox, p.229.) Fox confirms, for example, that the Web can be used to “[p]ublish[] text, images, software, or any other computer file to a potential audience of millions.” (Fox, p.229; *see also id.*, p.316 (“Proofread your pages for spelling and grammar errors. The page will be seen by millions of adoring fans around the world.”), p.198 (“Money makes the world go round, and the same can be said for the World Wide Web. A huge chunk of the Web’s growth is due to large corporations and enterprising entrepreneurs hoping to turn the Web into tomorrow’s most lucrative medium.”).)

113. An ordinarily skilled artisan would thus have been motivated to adapt

the hypertext system based on Shimizu into a web-based system, among other reasons, to allow node content to be viewed by a potentially enormous number of users via their web browsers. An ordinarily skilled artisan developing a system based on the proposed combination would have wanted to avoid being left behind in the growing popularity of the Web, and thus, would have been motivated to adapt the system to support web-based resources.

114. As an additional but related motivation to combine, allowing node content to be identified using URLs as disclosed in Fox would also have expanded the universe of content that could have been incorporated into nodes. This is because, under the proposed combination, nodes could have identified any content on the Internet accessible via a URL, and thus, could have drawn from a massive universe of available content resources beyond the content stored on the user's local computer or the user's local area network.

115. An ordinarily skilled artisan would have had at least a reasonable expectation of success. As noted, the Web was well-known and identifying resources using URLs, as described in Fox, was straightforward if not trivial. Fox itself confirms URLs were already being used to identify resources for purposes of sequential presentation systems, as confirmed by the example from Fox discussed above that uses industry standard HTML <META> elements. (Fox, p.316.) The

disclosures of Shimizu and Fox would have provided more than sufficient details to have enabled an ordinarily skilled artisan to implement the proposed combination as described, without undue experimentation.

District Court’s Claim Constructions of
“Show Structure of Nodes” and
Duration Requirement

116. Before leaving the discussion of claim 1[a], I note that my mapping of Shimizu is also consistent with claim construction rulings in prior litigations. I am informed that “**show structure of nodes**” was defined in a previous litigation involving the related ’451 patent as “*a structure that is arranged for the display of content by specifying one or more paths through a plurality of nodes.*” I am further informed that “**node**” was defined as “*an identifier of a resource that includes an address to the resource and the duration for which the resource’s content is to be presented by default.*” It is not necessary for me to opine on the correctness of these constructions because, as I will show below, the prior art readily satisfies each of them. As discussed above, the node and link information in Shimizu identifies a set of nodes, the content associated with each node, and the links that specify the order in which the nodes are displayed. It would have been obvious that entity information pointer slot **63** part of the node information structure in Shimizu provides “*an address to the resource*” because it provides a pointer identifying the location of the

content associated with the node. Ordinarily skilled artisans, in fact, frequently used the words “pointer” and “address” interchangeably. (See *Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.308 (“**pointer**... In programming and information processing, a variable that contains the memory location (address) of some data rather than the data itself.”).) Under the further combination with Fox, as explained above, each resource could have been identified by a Uniform Resource Locator (URL) pointing to the location of the resource (*e.g.* web page, image, etc.) on the Web. It was well-known that a URL provides another example of an address. (Fox, p. 13 (“A *URL* (or *Universal Resource Locator*) is the address of a document, file, or other Internet resource... The World Wide Web gives every article, newsgroup, database, computer, file, or other Internet fragment its very own URL.”).)

117. I note that the challenged patent itself does not specify any particular format or organization for the claimed “**structure**,” but rather, describes it as broadly referring to an arrangement containing a set of nodes. (’932, 2:61-65 (“This arrangement of resources, is defined as a ‘show structure’ or ‘structure,’ which contains a set of nodes. Each node represents a resource such as a web page, an executable file, or a data file stored at a predetermined location and accessed via its corresponding address.”).) The proposed combination similarly discloses a set that

includes an arrangement of nodes which are, as explained previously, defined by their node and link information.

118. I note that the prior litigation construction of “**node**” further requires that the node also include “*the duration for which the resource’s content is to be presented by default.*” I note that this concept is separately recited in claim 1[c] as the “**interactively variable duration information.**” Nevertheless, this requirement of the construction of “**node**” is rendered obvious by the proposed combination.

119. For example, Shimizu discloses a default display time (*i.e.*, a “predetermined” time or interval in which node content is displayed before proceeding to the next node):

It follows that, on the hypertext apparatus, it is necessary to set the items of link information with their order of display properly established beforehand. Only this procedure makes it possible to create an application program that may be displayed on the screen by reading, from primary or secondary storage and within a predetermined time, the node information held by the nodes that are linked sequentially starting from a particular node.

(Shimizu, 3:9-16.)

This makes it possible to build with ease an application program illustratively causing the display of the nodes to be switched in a specific order and at predetermined intervals.

(Shimizu, 5:11-14.) Shimizu also suggests the ability to vary the display duration between nodes by explaining that “[i]t is also possible to build with ease a system for arranging information that emphasizes temporal relations between nodes (e.g. chronological tables).” (Shimizu, 9:66-10:2.)

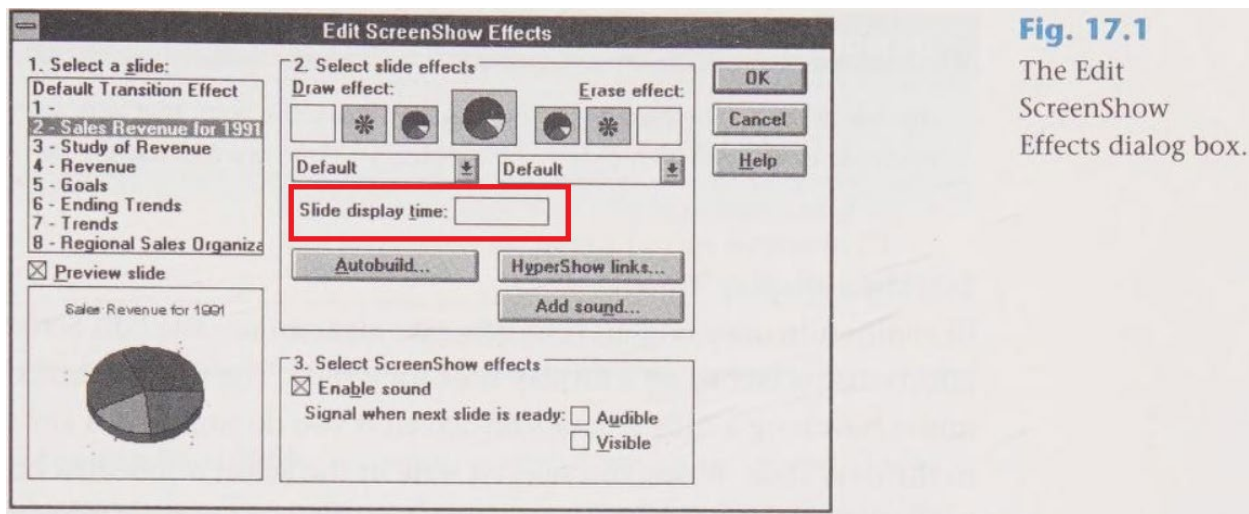
120. But Shimizu does not appear to disclose further details about how duration information is specified or stored within its system, or whether such intervals between nodes can be changed. To the extent it is determined that Shimizu alone does not disclose the durational requirement of the construction of “**node**,” it would have been obvious in further view of **Benedict**.

121. As explained in **Part V.A**, Benedict describes a software program known as Harvard Graphics that provided functionality similar, in some respects, to well-known business presentation programs such as Microsoft PowerPoint. (Benedict, p.1 (“Harvard Graphics for Windows is a program designed for creating business presentations.”).) Harvard Graphics allowed users to create a “presentation” including a series of slides. (Benedict, p.24 (“A *presentation* is a collection of slides stored in computer memory in a file on your hard disk.”) (italics in original).) Benedict also discloses features for viewing a presentation (referred to as a “ScreenShow”), for example, by displaying slides in a specified sequential order. “As you create and enhance presentations, you can use the ScreenShow

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

feature to evaluate your work. ScreenShows are on-screen presentations in which the program displays each slide in order.” (Benedict, p.19.) As noted, Benedict further discloses a feature known as “HyperShows” in which slides in a ScreenShow can include hypertext-like links allowing the user to switch to other slides in the ScreenShow by clicking on-screen links. (Benedict, pp.442-453.)

122. As relevant to the above-mentioned durational requirement, Benedict discloses additional features (known as “ScreenShow Effects”) that allow the user to customize the display of the presentation. (Benedict, pp.438-442.) As shown in Figure 17.1 of Benedict, one of these features allows the user to specify how long a particular slide in a presentation will be displayed before moving to the next slide:



(Benedict, p.439, Fig. 17.1 (red box added).) The portion in the red box, as shown above, allows the user to specify how long the selected slide will be displayed:

The display time determines how long a slide remains on-screen if you

do not press a key to move to the next slide...

The default for the time is blank because Harvard Graphics continues to display the slide until you press a key. You can set a display time, however, by typing a time in the Slide display time: text box. You can enter the time in seconds or in minutes and seconds separated by a colon. Entering 23, for example, sets the display time to 23 seconds; entering 1:30 sets the display time to 1 minute and 30 seconds.

(Benedict, p.440 (italics in original).) Benedict explains that, using this feature, the user can create “self-running” presentations in which slides advance automatically after a specified period, without requiring user action:

Creating Self-Running Presentations

Self-running presentations automate your ScreenShow by displaying the slides for a specified length of time before the next slide appears—pressing a key is not necessary. At the end of the presentation, a HyperShow link causes the ScreenShow to start over from the beginning. The presentation continues until you press the Escape key. To create a self-running presentation, you must set the display time for the slides and create the link that begins the presentation again.

(Benedict, p.453 (boldface in original).)

123. The user can also specify a default display time applicable to the presentation, without having to separately assign the time to each slide:

Setting Default Effects... When you select a default transition,

Harvard Graphics uses the same effect to draw and erase all slides.

[* * *]

When the [Edit ScreenShow Effects] dialog box appears, follow these steps to change the default effects and time display:

[* * *]

4. Click the Set display time: text box and type the default display time.

If you do not set a time, the slides remain on-screen until you press a key.

(Benedict, p.441-442, ¶4.)

124. *Manner, Rationale and Motivation to Combine (Shimizu and Fox with Benedict)*: It would have been obvious to combine Shimizu and Fox with Benedict, predictably resulting in the hypertext display system of the Shimizu-Fox combination with the ability to interactively vary, for each node, the duration of display for the node's content. Accordingly, the “**node**” in Shimizu under the proposed combination would have further included information specifying this user-specified duration. In a real-world implementation of the proposed combination, for example, the display time could have been specified within the node information structure **9** of Shimizu. I note that node information structure **9** in Shimizu also contains display procedure slot **62** (Fig. 6), which points to “a procedure program function used when the node in question is to be displayed on the screen.” (Shimizu,

6:52-55.) Shimizu thus contemplates that node information structure 9 may contain information about how the node will be displayed to the user, and as such, it would have been obvious that durational information could also have been included as part of node information structure 9. This durational amount could have been a default display time specified by the user, or a display time specific to the node in question, as explained above.

125. Benedict is analogous reference in the same field of techniques for providing organized or sequenced presentation of content, and would have been reasonably pertinent to a number of problems facing the inventor. For example, Benedict would have been reasonably pertinent to the problem of how to structure and organize information about nodes, in order to keep track of their associated content and the nodes to which they are linked, and how to implement display of node content for a specified duration based on user preference.

126. Shimizu, as explained above, already discloses the ability to display node content for a “predetermined” time or interval (Shimizu, 3:9-16, 5:11-14, 9:63-66). Shimizu also provides a suggestion to vary this predetermined duration between by explaining that “[i]t is also possible to build with ease a system for arranging information that emphasizes temporal relations between nodes (e.g. chronological tables).” (Shimizu, 9:66-10:2.) It would have been obvious to enhance Shimizu

based on Benedict to allow the user to specify the display time. An ordinarily skilled artisan would have been motivated to make this enhancement to improve user experience by providing greater customization of the duration that node content is displayed.

127. This would have provided clear benefits over a fixed-length duration that could not be modified by the user. An ordinarily skilled artisan (and even laypersons) would have readily understood the benefits of allowing the user to specify the display time. For example, nodes may have included content that was more complex thus justifying a longer display time to provide the user with more time to view and digest the content. Conversely, nodes with less or simpler content may have benefitted from a shorter display time, as this would have allowed the user to progress more quickly through nodes. In any case, an ordinarily skilled artisan would have been motivated to combine Shimizu with the “display time” teachings of Benedict to obtain the advantages of an interactively variable duration for display of node content.

128. An ordinarily skilled artisan would have had at least a reasonable expectation of success and could have implemented the proposed combination using conventional programming techniques. As noted, Shimizu itself already contemplates switching the display of nodes at “predetermined intervals” (Shimizu,

5:11-14), and it would have been trivial to adapt the node information structure in Shimizu to include user-provided information specifying the display time for the corresponding node. Shimizu itself clearly contemplates the ability to customize the display of nodes, and as explained above, node information structure **9** (Fig. 6) includes display procedure slot **62** which points to a program function or procedure used to display the node content, thus making clear that node information structure **9** can hold information about how the node will be displayed. (Shimizu, Fig. 6, 6:52-55.) The disclosures of Shimizu and Benedict would have provided more than sufficient detail to enable an ordinarily skilled artisan to implement the proposed combination as described, without undue experimentation.

(b) “without requiring user input, automatically accessing a plurality of said network accessible resources identified by a corresponding node; and” (Claim 1[b])

129. This limitation adds little to what was already discussed for claim 1[a]. Shimizu discloses that the system automatically accesses a plurality of resources, i.e., the content items associated with the nodes, for example, when the content resources associated with a node (“**a plurality of... resources identified by a corresponding node**”) are displayed:

FIG. 11 is a flowchart of steps in which the embodiment displays the contents of nodes in accordance with the order of their links. This

processing is started using the mouse to click the ‘PLAY’ button in the operation designating area **82** of the window **80** in FIG. 8. The contents of the linked nodes are displayed in the order set by the link information.

(Shimizu, 8:11-16.) Shimizu further discloses that, in order to display the node contents, the system uses the entity information pointer slot **63** discussed in the previous limitation to identify the location of the content resources, and then to access and retrieve it:

Step 107 accesses and reads the entity information linked by the entity information pointer of the node information structure designated by the pointer pointing to the acquired node. When the entity information (character string, image, sounds, motion pictures) is acquired from the node information entity management part 8, the content of the acquired information is reproduced and displayed in the node content display area 81 of the window 80 by applying the program which is associated with the pointer in the display procedure slot 62 of the node information structure 9.

(Shimizu, 8:49-59.) As explained for 1 [a], it would have been obvious based on Fox that the resources would have been “**network accessible**” resources. The proposed combination therefore renders obvious “**accessing a plurality of said network accessible resources identified by a corresponding node,**” as claimed. Under the

proposed further combination with Fox, as noted, the entity information pointer slot **63** would have included a URL associated with the text, image, sounds, or motion pictures on the Web to be displayed with the node. For example, if a URL pointed to a web page, the web page and any content resources on that page (*e.g.* images, videos, etc.), would have been accessed using known techniques, such as using a web browser.

130. Shimizu also confirms that the claimed step of accessing occurs **automatically**, and **without requiring user input**, because the access and subsequent display proceed from node-to-node without user intervention at predetermined intervals, based on the link information. (Shimizu, Fig. 11, (Steps 104-109), 8:67-9:4 (“In step **105**, the next node information is reproduced in accordance with the next link information. The processing above is carried out so that the node C is displayed followed by node B and node D, in accordance with the order of the link information.”), 5:11-14 (“This makes it possible to build with ease an application program illustratively causing the display of the nodes to be switched in a specific order and at predetermined intervals.”).) Shimizu does not disclose any user controls (*e.g.*, a “Next” button) for advancing to the next node. It thus would have been obvious that the content resources would have been accessed and presented on the user’s computer in continuous sequential fashion, one after another,

without requiring any input from the user beyond a single initial click of a PLAY button that serves to commence the node display process. As I will explain below, the claim does not exclude a system in which the automatic sequential display process is commenced in response to a user pressing a PLAY button.

131. Benedict similarly discloses accessing “**automatically**” and “**without requiring user input,**” because ScreenShow presentations can be “self-running” and thus proceed automatically:

Creating Self-Running Presentations

Self-running presentations automate your ScreenShow by displaying the slides for a specified length of time before the next slide appears—pressing a key is not necessary. At the end of the presentation, a HyperShow link causes the ScreenShow to start over from the beginning. The presentation continues until you press the Escape key. To create a self-running presentation, you must set the display time for the slides and create the link that begins the presentation again.

(Benedict, p.453.)

132. Fox also similarly discloses the ability to access resources automatically, as the “REFRESH” technique discussed above automatically retrieves the next page after the elapsed period of time. (Fox, p.315 (“A Web page can be programmed to tell a server to reload its data after a given amount of time.

The server waits and automatically gives the Web page the new information.”); *id.*, p.316 (“To load up a different document after a set amount of seconds, simply specify a full URL using the URL attribute.”).)

133. Under the proposed combination of Shimizu, Benedict and Fox, as noted, the user would initially activate a “PLAY” button or other control to initiate the automatic sequential display of the content resources identified by the nodes. (Shimizu, 8:17-19, Fig. 11 (Step 101).) This presents no obstacle to obviousness because the actual step of **“accessing a plurality of said network accessible resources identified by a corresponding node,”** is separate from user activation of a PLAY button – the step is performed automatically by the software. The “PLAY” button or other user control is merely an unclaimed prerequisite user step that does not perform the actual claimed step of accessing the claimed plurality of resources. This is also consistent with the challenged patent, which similarly discloses a “Play button” functionality for initiating the automatic presentation of nodes. (’932, 16:25-26 (“The Play button begins the presentation of information.”).) The ’932 specification further explains that “[a] triggering event, such as a mouse click on a single ‘link’ commences the presentation of a series of resources instead of one resource at a time.” (’932, 2:59-61.) I note that the district court’s claim construction order expressly adopted Patent Owner’s argument that **“without**

98

requiring user input” did not exclude user action that triggers the commencement of the show. Accordingly, an ordinarily skilled artisan would not have interpreted the “**automatically**” and “**without requiring user input**” limitations as precluding any type of input that causes initiation of node display.

134. I further note that prerequisite user steps, including the initial activation of a “PLAY” button, would not perform the actual claimed step of accessing “**a plurality**” said resources associated with the plurality of nodes, because advancement to sequential nodes (which would have been required to access and display the claimed “**plurality**” of resources) would have taken place automatically and without requiring any user input. Finally, I note that even if the claim required display of resources from start-to-finish without an initial user step in the causal chain (such as clicking a “PLAY” button), this would still have been obvious. As noted, Benedict above expressly discloses that a slide presentation can be configured, after its display is complete, to automatically and repeatedly replay itself. (Benedict, p.453 (“At the end of the presentation, a HyperShow link causes the ScreenShow to start over from the beginning. The presentation continues until you press the Escape key.”).) As such, the second, third, etc., runs through the set of nodes under the proposed combination – any one of which would have been sufficient to satisfy the claim – are even further removed from any previous user

steps in the causal chain including the initial press of a “PLAY” button.

- (c) **“causing content from said network accessible resources corresponding to each of said accessed resources to be automatically accessed and presented through said user's computer in accordance with said show structure such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed,” (Claim 1[c])**

135. As just explained for claim 1[b], the resources associated with each node (*e.g.*, character strings, images, sounds, motion pictures, etc.) are automatically accessed and retrieved. Because these resources (*e.g.* files, web pages, etc.) contain the underlying content, by accessing and retrieving these the resources, **“content from said network accessible resources corresponding to each of said accessed resources”** is also automatically accessed. The content is then automatically presented through the user’s computer. (Shimizu, 8:11-16, 8:49-59; Benedict, p.435 (“Self-running presentations automate your ScreenShow by displaying the slides for a specified length of time before the next slide appears—pressing a key is not necessary.”); Fox, p.316) The content is automatically displayed **“in accordance with said show structure,”** because it is presented in the sequence specified by the corresponding link information for the nodes. (Shimizu, 8:67-9:4, 5:11-14.)

136. Claim 1[c] continues by reciting that **“during said presentation of at**

least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.” This limitation is obvious based on Benedict (discussed above) and Wolfe.

137. As explained in **Part V**, this claimed feature recites nothing more than a well-known technique, commonly referred to as “pre-fetching” or “pre-loading,” in which a computer fetches or loads data before the point in time when the data is actually needed. When the computer later needs that requested data, it can utilize it immediately because it has already been fetched or loaded.

138. Pre-loading was also a commonly used technique for reducing delays associated with presenting information to a user. For example, Benedict discloses a feature in which Harvard Graphics can prepare for the display of a subsequent slide during the display of a current slide, and notify the user when the next slide is ready for display:

Signalling [sic] for the Next Slide. Depending on the power and speed of your computer, Harvard Graphics may need time to prepare the next slide in the presentation before displaying it. The Signal when next slide is ready: options in the Edit ScreenShow Effects dialog box enable you to activate an audible signal (a beep) and visual signal (a small arrow in the bottom right corner of your screen) to indicate when the next slide is ready.

(Benedict, p.441 (boldface in original).) As shown, Benedict confirms that “Harvard Graphics may need time to prepare the next slide in the presentation before displaying it” (*id.*), but by preparing the next slide during the time the user is viewing the current slide, the next slide can be displayed when needed without unnecessary delay.

139. Benedict renders this aspect of claim 1[c] obvious because during the presentation of the content resources of a current slide, the resources in at least a subsequent slide are being accessed in order to “prepare the next slide in the presentation before displaying it.” (*Id.*) An ordinarily skilled artisan would have understood that the act of “accessing” information is a broad one that encompasses opening a file and reading information from it, among other acts. (*See Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.7 (“**access** To obtain entry into, or to locate, read into memory, and make ready for, some operation.”).) An ordinarily skilled artisan would thus have appreciated that resources are “**being accessed**” when, for example, the computer is retrieving them from storage and loading them into memory for use.

140. In the case of Benedict, in order to prepare the next slide for display, the software would have accessed resources associated with that next slide, such as by reading those resources into memory in order to prepare the display of the next

slide. Under the proposed combination, therefore, it would have been obvious that **“during said presentation of at least some of said content from said accessed resources,”** *i.e.*, during presentation of content resources associated with the currently displayed node, **“at least some of said other network accessible resources are being accessed,”** *i.e.*, content resources associated with the next node are being accessed (such as by retrieving them from the Internet via their URLs) in order to prepare the next node for display. An ordinarily skilled artisan would have been motivated to implement this technique to improve the performance of the system and user experience by eliminating or reducing the amount of delay before the display of the next node.

141. Benedict does not disclose details about how the next slide preparation technique works, so the extent Benedict is faulted for any lack of technical detail, I have also cited to **Wolfe**, which discloses more details about a similar technique for pre-loading information while the user is viewing other information.

142. Wolfe explains that a sequence of documents can be identified for the user, for example, by using the results of a search query. (Wolfe, 1:53-57 (“Whatever querying technique is used, a search engine uses the query [to] define a class of documents within the database, and locates each of the documents within the class. The user then browses the search results by ‘paging through’ one or more

of the search documents.”).) But Wolfe explains that “today’s database systems are plagued by the problem of slow document retrieval.”

Consequently, today’s database systems are plagued by the problem of slow document retrieval. After a query is processed and search documents are identified, the user begins browsing the search results by studying the first view (or screen of information) from one of the search documents. The user then either “pages-down” to the next view within the same document, or moves to another search document. Each of these moves requires that new information be retrieved from the database before it can be displayed on the screen. Since the retrieval time is substantial... the database system is slow in responding to the user's request for the information.

(Wolfe, 2:1-12.) Wolfe addresses this issue through a technique in which, while the user views certain information, other information the user will likely to need at a later point is preloaded:

This need can be satisfied by effectively utilizing the time the user spends studying the information on the display screen. In a database system (or document retrieval system) in one embodiment of the present invention, information that the user is likely to eventually request is preloaded into memory while the user is viewing other information. The present invention takes advantage of the fact that it is possible to accurately predict the information that the user will eventually request be shown on the display. The present invention also

takes advantage of the fact that the time that the user spends viewing displayed information is often sufficient to advantageously preload a substantial amount of information.

(Wolfe, 2:23-35.) Wolfe further explains:

In the systems of the prior art, when the database system receives the command to display a different view, the requested view must be loaded from the database **30** before it can be displayed on the monitor **27**. Since retrieving information from the database is time-consuming, this loading process is undesirably slow. But in a system employing the present invention, the time required to respond to the user's request for a different view (the “requested view”) is reduced by taking advantage of the fact that it is often possible to predict the requested view before the user actually requests it. In the present invention, the view(s) that the user is likely to next request are preloaded while the user is reading the displayed view.

(Wolfe, 5:47-59.) Although an exemplary embodiment in Wolfe involves an on-line legal database, Wolfe makes clear that his techniques are broadly applicable to any type of documents that can be retrieved and presented to the user. (Wolfe, 4:20-23 (“But a ‘document’ need not only be text—it could be a picture, video clip, or information in another format, such as might be available in a multimedia encyclopedia.”).) Wolfe also makes clear that his techniques are applicable to both single-user databases as well as networked databases accessible by multiple users

simultaneously. (Wolfe, 3:60-4:8.)

143. *Rationale and Motivation to Combine (Shimizu, Fox, and Benedict with Wolfe)*: It would have been obvious to combine with Wolfe, predictably resulting in the Shimizu-Fox-Benedict system with the further ability to access “**at least some of said other network accessible resources,**” *i.e.*, content resources on the Internet associated with the next node in the show structure, “**during said presentation of at least some of said content from said accessed resources,**” *i.e.*, during display of content resources for the current node. For example, during the display of the current node to the user, the system would have retrieved the resources (*e.g.*, Web pages, images, etc.) associated with the next node over the Internet.

144. Wolfe is an analogous reference in the field of content retrieval over networks, and would have been pertinent to a number of problems facing the inventor. For example, Wolfe would have been pertinent to implementing techniques for accessing content resources during the display of other content resources, including benefits of incorporating such techniques into the invention.

145. An ordinarily skilled artisan would have been motivated to combine to improve user experience, by reducing or eliminating the delays associated with accessing and displaying the next node’s content resources. Benedict supports this motivation by explaining “Harvard Graphics may need time to prepare the next slide

in the presentation before displaying it” (Benedict, p.441), and that same observation would have applied to any system that presents information sequentially to the user – including the proposed combination. Wolfe supports this motivation by explaining, as noted, that the “undesirably slow” loading process can be reduced by preloading subsequently-needed information during the display. (Wolfe, 5:47-59.) Wolfe also explains that his preloading technique is particularly appropriate in a scenario in which the system can predict what information the user will subsequently require. (Wolfe, 2:23-35.) The proposed combination presents precisely that scenario because nodes are presented to the user in a predetermined (and thus predictable) sequence based on the link information. Wolfe thus provides a clear advantage because, during the time when the user is reviewing the current node, the system can access network accessible resources associated with the next node.

146. Nor does Wolfe in any way “teach away” from the proposed combination. A passage in Wolfe explains that his preloading technique is potentially more effective to single-user databases such as those in which data is loaded from local storage such as a CD-ROM, because in a single-user database, no competing demands by other users exist. (Wolfe, 9:13-18.) But in a many-user or multiple-user database (such as a database accessed over a network), Wolfe asserts, simultaneous requests may exist from multiple users resulting in the invention being

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

less effective because the database may not be in an idle state when the requests are received. (Wolfe, 9:19-26.) An ordinarily skilled artisan would not have interpreted these statements as discouraging the use of preloading with a network accessible database such as a many-user or multiple-user database, as Wolfe acknowledges that even with these types of databases, “where the database is not as busy, the present invention can be effective in reducing response times to users’ requests for information.” (Wolfe, 9:27-29.) It would have been obvious to an ordinarily skilled artisan that the performance increase achieved by preloading would have depended on a number of additional factors, including the amount of time the current node is being displayed before advancement to the next node (as a longer duration would have provided more time for preloading), as well as the network speed and latency.

147. Wolfe’s comments also appear to pertain to highly centralized databases simultaneously accessed by a large number of users. But under the proposed combination, content resources associated with nodes (identified by URLs) could have been located across multiple servers on the Web, thus avoiding the centralization concerns of Wolfe. And even in a system involving a highly centralized database or server, an ordinarily skilled artisan would have understood that the preloading will still achieve benefits because – regardless of the response time or congestion at the database or server – the requested information would have

been provided earlier because it was requested at an earlier point in time.

148. An ordinarily skilled artisan would have had at least a reasonable expectation of success, and could have implemented the proposed combination using conventional programming techniques. Techniques for loading content resources over a network (such as the Internet) were well-known. The ability to access resources associated with a subsequent node, during display of a current node, could have been implemented using conventional techniques. The disclosures of Wolfe and Benedict would have provided more than sufficient details to have enabled an ordinarily skilled artisan to implement the preloading features I am citing for this claim limitation, without undue experimentation.

(d) “wherein said step of creating further comprises the step of providing interactively variable duration information, representing the duration that said content corresponding to said resource is presented so as to enable a user to vary said duration.” (Claim 1[d])

149. Shimizu also discloses that the step of creating the show structure of nodes also comprises providing duration information (*i.e.* a “predetermined interval” for the display of the content associated with a node):

As described, the hypertext apparatus of the invention offers a user interface capable of putting into a desired order the links between the nodes to be managed and of manipulating the established link order.

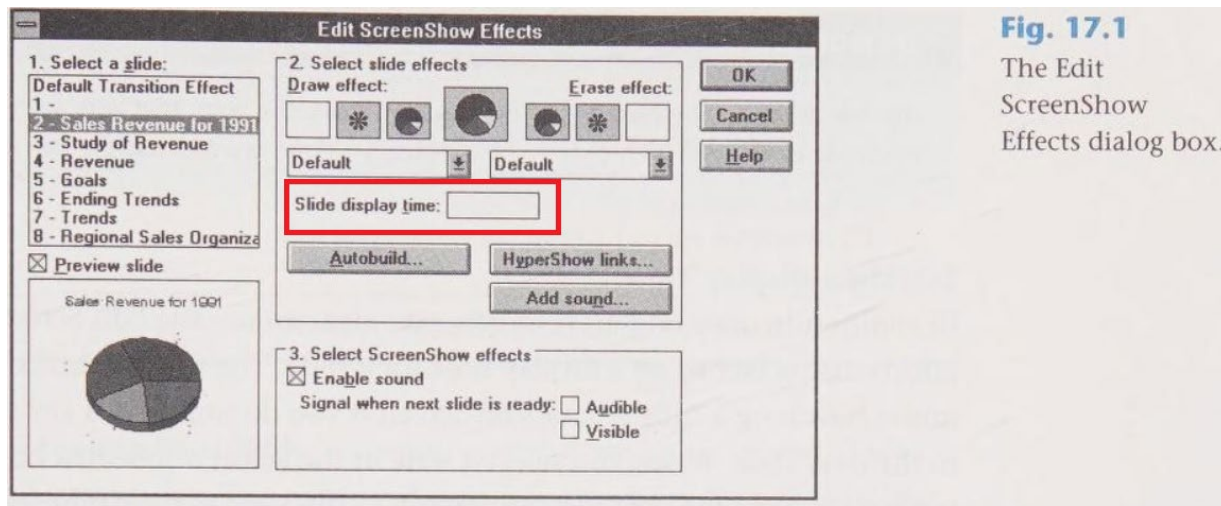
This makes it easy to create illustratively an application program allowing the display of node contents to be switched in a desired order and at **predetermined intervals** on the hypertext apparatus.”

(Shimizu, 9:60-66; *see also id.*, 5:11-14 (“This makes it possible to build with ease an application program illustratively causing the display of the nodes to be switched in a specific order and at predetermined intervals.”).)

150. As explained for claim 1[a], Shimizu does not disclose further details about the predetermined interval the switching the display from one node to the next, and does not expressly disclose that the duration information is “**interactively variable**” or that the content is displayed “**so as to enable a user to vary said duration.**” But this would have been obvious in further view of **Benedict**.

151. As explained for claim 1[a], under the combination of Shimizu with Benedict, each node would have included a display time that could have been interactively varied by the user – under two separate theories. First, Benedict discloses that the user can change the display time for a node using the ScreenShow Effects feature. (Benedict, *e.g.*, p.453 (“You set links and display times in the Edit ScreenShow Effects dialog box.”), p.440 (“You can set a display time... by typing a time in the Slide display time: text box.”).)

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2



(Benedict, p.439, Fig. 17.1 (red box added).) As Benedict explains:

You can enter the time in seconds or in minutes and seconds separated by a colon. Entering 23, for example, sets the display time to 23 seconds; entering 1:30 sets the display time to 1 minute and 30 seconds.

(Benedict, p.440 (italics in original).) As noted previously, this technique can be used to edit the display time for each slide individually, or for all slides. Benedict thus discloses that a user can interactively (through the user interface) vary the display time for a slide. Under the proposed combination of Shimizu and Benedict, therefore, the duration information for a node (specifying how long its content will be displayed) can be interactively varied by a user – and if desired, on a node-by-node basis. For purposes of this first way, the “user” may be a user creating or editing the presentation, who could have been the same user who would also be viewing the presentation.

152. As a second and independent way of satisfying the “**interactively variable duration information**” limitation, I am informed that in the pending litigation, Patent Owner has taken the position that a user can interactively vary the duration information through a user interface control that allows the user to skip or advance to the next item. I am further informed that Petitioner does not agree that such functionality satisfies the requirement of interactively variable duration information. Nevertheless, in the event the claim were construed to encompass this functionality, the prior art discloses and renders it obvious. More specifically, Benedict discloses that a user can vary the display time by pressing a button during a ScreenShow displaying a current slide to advance to the next slide without having to wait the full display time: “The display time determines how long a slide remains on-screen if you do not press a key to move to the next slide.” (Benedict, p.440; *see also id.*, p.443 (“Next slide... Moves to the next slide in the presentation”), *id.* (“In addition to keys and buttons, you can define a link that determines the length of the slide display time.”).) The user under the proposed combination can thus vary (*i.e.* shorten) the display duration by immediately advancing to the next node before expiration of the node’s specified duration. I note that for purposes of this second way, the “user” could include the user viewing the presentation, and as noted, this could be the same user who would be creating or editing it.

153. Under the proposed combination, therefore, a user can interactively vary the duration within which a corresponding content to a node is being displayed, either **(1)** by allowing a user to interactively change the “display time” for one or more nodes through a user interface or **(2)** by allowing a user to press a button during display of node content to shorten/truncate the display time, to the extent it is determined that the claim encompasses such functionality.

154. I am informed that in a previous litigation involving the related '451 patent, a district court construed “**interactively variable duration information**” as *“a parameter specifying how long a content is to be displayed by default before a subsequent content is displayed, where the viewer of the content can change the parameter.”* This construction is also satisfied by the proposed combination. In situation **(1)** discussed in the previous paragraph, the parameter corresponds to the duration for the display of node content, *i.e.*, a default display time (“predetermined interval”) in Shimizu before advancing to the next node, which the viewer can change as needed through a user interface such as Edit ScreenShow Effects dialog box in Benedict. It would have been obvious that the “predetermined” interval in Shimizu is a default value, because the intervals are “predetermined” and nothing in Shimizu suggests that they were selected by the user. In situation **(2)**, to the extent it is determined that the claim encompasses such functionality, the parameter is the

default amount of time the software waits when actually displaying a particular node contents before advancing to the next node (based on the established display time) – which the user can shorten by taking action (such as pressing a button) to advance to the next node before expiration of the display time, as explained.

155. This is also consistent with Fox, which confirms that the ability to change the display time could have been accomplished using any number standard web-based techniques such as a text field to solicit the display time from the user (Fox, pp.396-397), and immediately switching to the next node could have been accomplished using a standard URL or hyperlink to implement a “Next” button or link (Fox, p.238 (“The [Next] button loads up the next page in the series, and the [Previous] button, naturally, loads up the previous page.”).) The system built based on the proposed combination, therefore, would have been further adapted to include a user interface feature for allowing the user to specify the time for displaying the content of individual nodes (*e.g.*, as a text box or other suitable control), and a feature for allowing a user to advance to the next node without having to wait the full display time. Under either theory, therefore, the duration could have been interactively varied by the user/viewer of the content. The rationale and motivation to combine Shimizu, Benedict, and Fox was provided for claim 1 [a] and applies here.

2. Claim 2

156. Claim 2 depends from claim 1, and further recites “[t]he **method in accordance with claim 1, wherein said step of providing interactively variable duration information further comprises the step of providing interactively variable duration information for each of said accessed resources, representing the duration within which a corresponding content to said resource is presented.**” The prior art renders obvious the claimed “**interactively variable duration information**” for the reasons stated for claim 1[d] above.

157. Claim 2 further requires providing “**interactively variable duration information**” for “**each of said accessed resources,**” which adds nothing of patentable significance. The duration information under the proposed combination would have been applicable to each of the accessed resources because it would have specified how long its corresponding content would have been displayed before proceeding to the next node.

158. Claim 2 does not appear to require providing *multiple different* durations for the accessed resources. In other words, a duration is provided “**for each of said accessed resources,**” under the proposed combination, by providing a duration applicable to all nodes and their associated resources. But even if the claim were interpreted more narrowly, as explained for claim 1[a], Benedict also discloses

the ability to specify a display time separately for each slide, *e.g.*, on a slide-by-slide basis. (Benedict, p.439, ¶3 (“To see the effects for a particular slide, select the slide from the Select a slide: list box.”); *id.*, 440 (“In addition to draw and erase effects, you can also use the Edit ScreenShow Effects dialog box to set a display time for a slide.”).) It would have been obvious to provide separate interactively variable duration information for each accessed resource, to the extent the claim were construed to impose such a requirement. An ordinarily skilled artisan would have been motivated to provide this feature, as explained for claim 1[a] and 1[d], to provide greater flexibility. As explained, some nodes may have included complex content warranting a longer display time whereas other nodes may have included simpler content requiring a shorter display time.

3. Claims 3-6

159. For purposes of my analysis, these claims are closely related and thus can be addressed together. Claim 3 depends from claim 1, and further recites “**wherein one or more of said nodes comprises a URL.**” Claim 4 depends from claim 1, and further recites “**wherein each of said nodes of said show structure of nodes identifies a network accessible resource located on a remote computer..**” Claim 5 depends from claim 1, and further recites “**wherein one or more of said**

nodes identifies a resource accessible via the Internet.”⁵ Claim 6 depends from claim 1, and further recites “**wherein one or more of said nodes identifies a resource accessible via an Internet website.**”

160. As explained for claim 1[a], each node under the proposed combination “**comprises a URL**” (claim 3) which could have been stored, for example, in the entity information pointer slot **63** of the node information in Shimizu. Each of these URLs also identifies “**a network accessible resource located on a remote computer**” (claim 4), “**a resource accessible via the Internet**” (claim 5), and “**a resource accessible via an Internet website**” (claim 6) because it identifies a Web resource (*e.g.* web page, image, etc., accessible from a website) located on a remote web server computer via the Internet. (Fox, *e.g.*, p.14 (“The second part of most URLs is the full Internet address of the machine where the file is located.”).)

4. Claims 7-9

161. For purposes of my analysis, these claims are closely related and thus can be addressed together. Claim 7 depends from claim 1, and further recites “**wherein one or more of said nodes identifies video content.**” Claim 8 depends

⁵ A Certificate of Correction signed on June 9, 2015 corrects claim 5 to change the word “identities” between the words “**nodes**” and “**a**” to “**identifies.**”

from claim 1, and further recites “**wherein one or more of said nodes identifies audio content.**” Claim 9 depends from claim 1, and further recites “**wherein one or more of said nodes identifies displayable content.**”

162. As explained for claim 1[a], Shimizu explains that each node can identify content such as “a character string, image information, motion pictures or sounds.” (Shimizu, 5:55-59; *see also id.*, 8:52-59, 1:23-29.) Fox similarly discloses that Web resources can include video and audio content. (Fox, p.45 (“The Web combines graphics, video, sound, text, hypertext, and more.”).) It thus would have been obvious that one or more of the nodes would have identified video and/or audio content, both of which are examples of displayable content.

5. Claim 10

163. Claim 10 depends from claim 1, and further recites “[t]he **method in accordance with claim 1, wherein said show structure is displayed to the user.**” Shimizu and Benedict both disclose this feature. With respect to Shimizu, Figure 8 shows a user interface screen for presenting node content to the user:

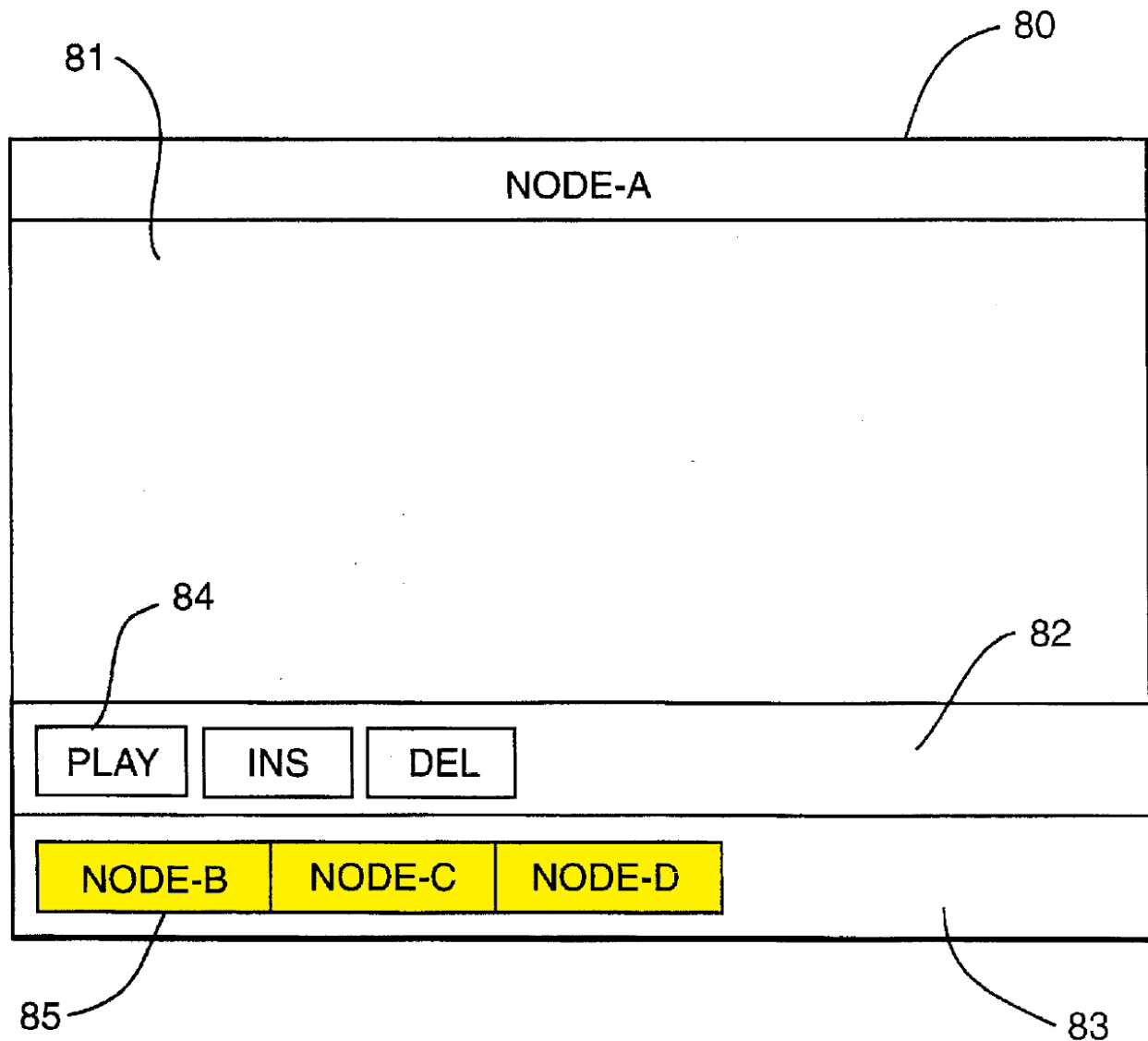


FIG. 8

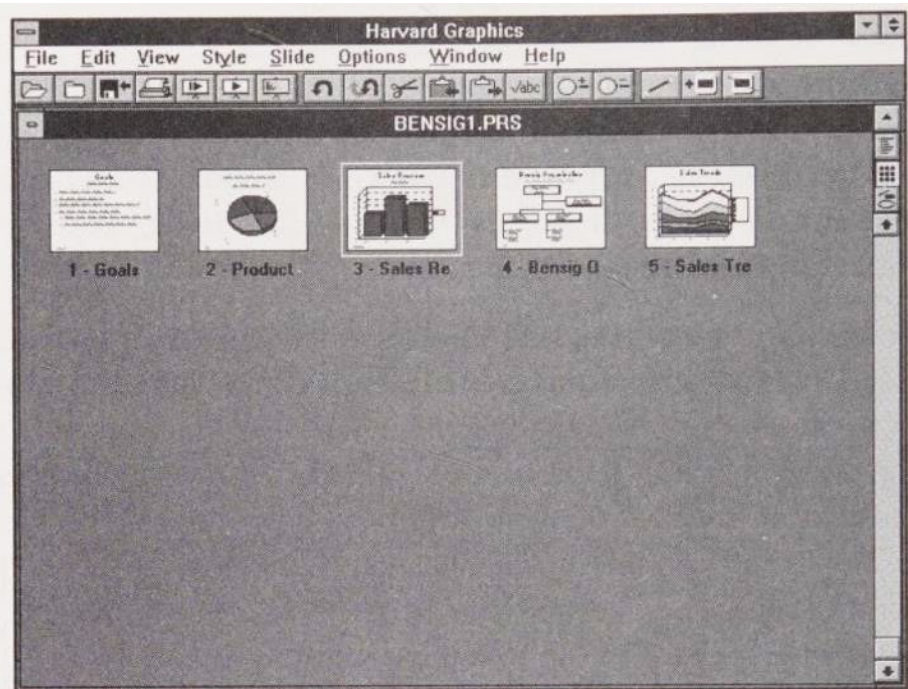
(Shimizu, Fig. 8 (highlighting added), 7:12-14 (“FIG. 8 is a view of a typical user interface screen display provided by an application program running on the embodiment.”).) Figure 8 includes the display of a current node (NODE-A) and display area **83** that “displays, in a left-justified manner and according to the link

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

information order, icons **85** representing each of the nodes linked starting from the current node. The icons **85** show in a list format the links connected to the current node and the order in which these links are established.” (Shimizu, 7:30-35.) As a result, “[t]he link information of the node A indicates that the node A is linked to the nodes B, C and D, in that order.” (Shimizu, 7:37-38.) Under the proposed combination, therefore, it would have been obvious that “**said show structure is displayed to said user,**” because icons **85** show icons for each node and the order in which they are linked.

164. This functionality is also separately disclosed by a similar feature in Benedict, *i.e.*, a “Slide Sorter” displays an icon for each slide in a presentation:

Fig. 2.18
The Slide Sorter
window.



(Benedict, p.44, Fig. 2.18; *see also id.*, p.43 (“Figure 2.18 shows the Slide Sorter displaying five slides. The Slide Sorter represents these slides as scaled down images.”).) Like Shimizu, Benedict discloses the display icons for each slide in the presentation, numbered based on the order in which they will appear. This provides another way in which it would have been obvious that **“said show structure is presented to said user.”**

6. Claim 11

165. Claim 11 depends from claim 1, and further recites **“[t]he method in accordance with claim 1, wherein said interactively variable duration information is displayed to the user.”** As explained for claim 1, Benedict discloses an Edit ScreenShow Effects user interface that displays a text box for the user to set or change the display time for one or more slides. (Benedict, p.439, Fig. 17.1; *see also id.*, p.440 (“You can set a display time... by typing a time in the Slide display time: text box.”).) It therefore would have been obvious that the **“interactively variable duration information is displayed to the user”** because the display time would have been displayed to the user, for example, in a text box when the user enters or edits the display time.

7. Claims 14 and 16

166. Claim 14 depends from claim 1, and further recites **“[t]he method in**

accordance with claim 1, further comprising the step of providing control buttons to enable the user to vary said duration that said content is being presented.” Relatedly, claim 16 depends from claim 14 and recites that **“said control buttons provide for skipping of said content being presented.”** These limitations are disclosed and rendered obvious by Benedict and Fox.

167. Benedict discloses that, during presentation of a HyperShow, on-screen control buttons can appear for controlling the display of content, such as navigating to different slides in the presentation:

Table 17.3 HyperShow Destinations	
Destination	Result
Next slide	Moves to the next slide in the presentation
Previous slide	Moves to the previous slide in the presentation
First slide	Moves to the first slide in the presentation
Last slide	Moves to the last slide in the presentation
Back up	Moves to the last slide shown on-screen
Ignore	Ignores the key so that nothing happens when you press a key
Launch application	Starts a Windows application from within the ScreenShow
Play sound file	Plays a sound file within the ScreenShow
Stop sound files	Stops sound files which were started from a previous HyperShow link

(Benedict, p.443.) As explained previously, users can also define their own

HyperShow links to jump to other parts of the presentation. (Benedict, pp.444-49.)

These links take the form of buttons. (Benedict, p.446 (“You create HyperShow buttons in the Slide Editor.”).)

168. Fox also similarly discloses, as a best practice to maintain a consistent user interface for navigating between web pages, navigation buttons for controlling the display of web pages:

The best way of achieving consistency is to use a menu bar, or a set of uniform links. You can include these navigation controls at the top or bottom of every page. For instance, many Web pages use the following links at the bottom of each document:

[Home] [Next] [Previous]

If a user clicks on [Home], then the home page appears. The [Next] button loads up the next page in the series, and the [Previous] button, naturally, loads up the previous page.

(Fox, p.238.) It would therefore have been obvious under the proposed combination, to provide control buttons to enable the user to vary the duration that the content is being presented, such as the “next” buttons as described in Benedict and Fox. These buttons would have enabled the user to **“vary said duration that said content is being presented”** (claim 14) by allowing the user to exercise control over the timing of the advancement of the show. For example, a “next” control button would have

allowed “**skipping of said content being presented**” (claim 16), by allowing the user to skip the current node and immediately advance to a subsequent node without having to wait for expiration of the duration for automatic advancement.

8. Claim 17

169. Claim 17 depends from claim 1, and further recites “[t]**he method in accordance with claim 1, wherein said computer is a mobile device.**” This claim adds nothing of patentability, as it would have been obvious and basic knowledge to any ordinarily skilled artisan that the user computer of claim 1 (“**said computer**”) could have been a mobile device, such as a laptop or other mobile computer.

170. As was well-known to ordinarily skilled artisans and even laypersons, mobile devices such as laptop computers were widely available and commercially popular by the early 1990s, several years before the priority date. (*See* EX1016 (Hollingsworth), 1:14-17 (“Recent years have seen a proliferation of so-called ‘laptop’ computers, that is, personal computers so reduced in size and weight as to be conveniently carried and employed while traveling or the like.”).) Laptop and mobile computers were not niche products, either; they were sold by household names such as Apple Computer (PowerBook), IBM (ThinkPad), Compaq Computer Corporation (which actually began selling portable PCs by 1983), among many others. Mobile computers by the mid-1990s, moreover, could have performed the

same functions as desktop computers including communicating over networks such the Internet. (EX1017 (Steele), 5:29-32 (“The client machine **10** may preferably be an intelligent computer system, such as an IBM PS/2 computer, an IBM ThinkPad laptop computer, or an IBM RISC System 6000 workstation.”).) It would have been obvious that the user’s computer could have been a mobile device such as a laptop or notebook computer, which could have performed all the functions of the user computer in claim 1 under the proposed combination. An ordinarily skilled artisan would have been motivated to enable the proposed combination to work with a mobile computing device because such a device would have been portable and thus more convenient for the user.

9. Claim 18

171. Claim 18 depends from claim 1, and further recites “[t]he **method in accordance with claim 1, wherein said content is presented through said user's computer using a web browser.**” As explained for claim 1, under the proposed combination, content resources are resources on the Web identified by URLs, such as web pages, images, etc. It would have been obvious to display that content through a web browser, as a browser was the standard software tool for displaying content from the Web.

10. Claim 19

172. Claim 19 depends from claim 1, and further recites “[t]he method in accordance with claim 1, further comprising the step of altering the show structure in response to user action.” Shimizu and Benedict disclose this feature in multiple ways. With respect to Shimizu, it discloses a DEL button (Fig. 8) that the user can activate to delete a node. (Shimizu, Fig. 8, 7:25-28 (“...the DEL button is used to delete a node.”).) Shimizu also discloses an INS button (Fig. 8) that allows the user to insert a new node at a particular position. (Shimizu, Fig. 8, 7:26-27, 9:5-59.) And Shimizu discloses a feature allowing the user (using a computer mouse) to change the order of nodes. (Shimizu, 7:40-48.) Similarly, Benedict discloses a Slide Sorter feature that allows the user to delete, add, or reorder slides in a presentation. (Benedict, pp.251-252.) Under the proposed combination, therefore, it would have been obvious to “alter[] the show structure in response to user action,” such as user action to delete, add, or reorder one or more nodes, each of which would have resulted in a change to the show structure.

11. Claim 20

173. Claim 20 depends from claim 1, and further recites “[t]he method in accordance with claim 1, wherein said show structure includes a sequential path of nodes.” As explained for claim 1[a], the show structure under the proposed

combination includes at least a sequential path of nodes. (Shimizu, *e.g.*, 7:62-65 (“Specifically, the content of the node A is displayed first followed by the contents of the nodes C, B and D, in that order, inside the node content display area **81** of the window **80**.”), 9:1-4; *see also* Benedict, p.19 (“ScreenShows are on-screen presentations in which the program displays each slide in order.”).)

12. Claim 21

174. Claim 21 depends from claim 1, and further recites “[t]he **method in accordance with claim 1, wherein said show structure includes a concurrent path of nodes.**” Benedict and Fox make clear that the nodes in the proposed combination need not be linked solely in single sequential fashion – there could be multiple different paths connecting the nodes. As explained in Benedict, each slide can include hyperlinks to jump to other slides in the presentation:

HyperShow goes beyond the special effects of ScreenShow in presenting information. HyperShow keys and buttons enable you to control the flow of your presentation spontaneously, without editing the original slide order. With HyperShow links, you can jump back to a previous slide to reinforce a point, for instance, or you can skip ahead to slides that are more relevant to an idea or question from the audience. The capability of viewing slides in any sequence turns an order-driven presentation into a presentation driven by information. Best of all, you can tailor the information to accommodate the interests and focus of

your audience.

(Benedict, pp.435-36.)

175. The capability to view nodes (which in Benedict are represented as slides) “in any sequence” was basic and well-known functionality of hypertext systems such as the Web. It was well-known that each page in a collection of Web pages could have had hyperlinks to other pages in the collection, allowing the user to navigate the set of pages out of order. For example, Fox discloses a NYC Nite Life web site having five pages, each containing a hyperlink to the other four pages:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

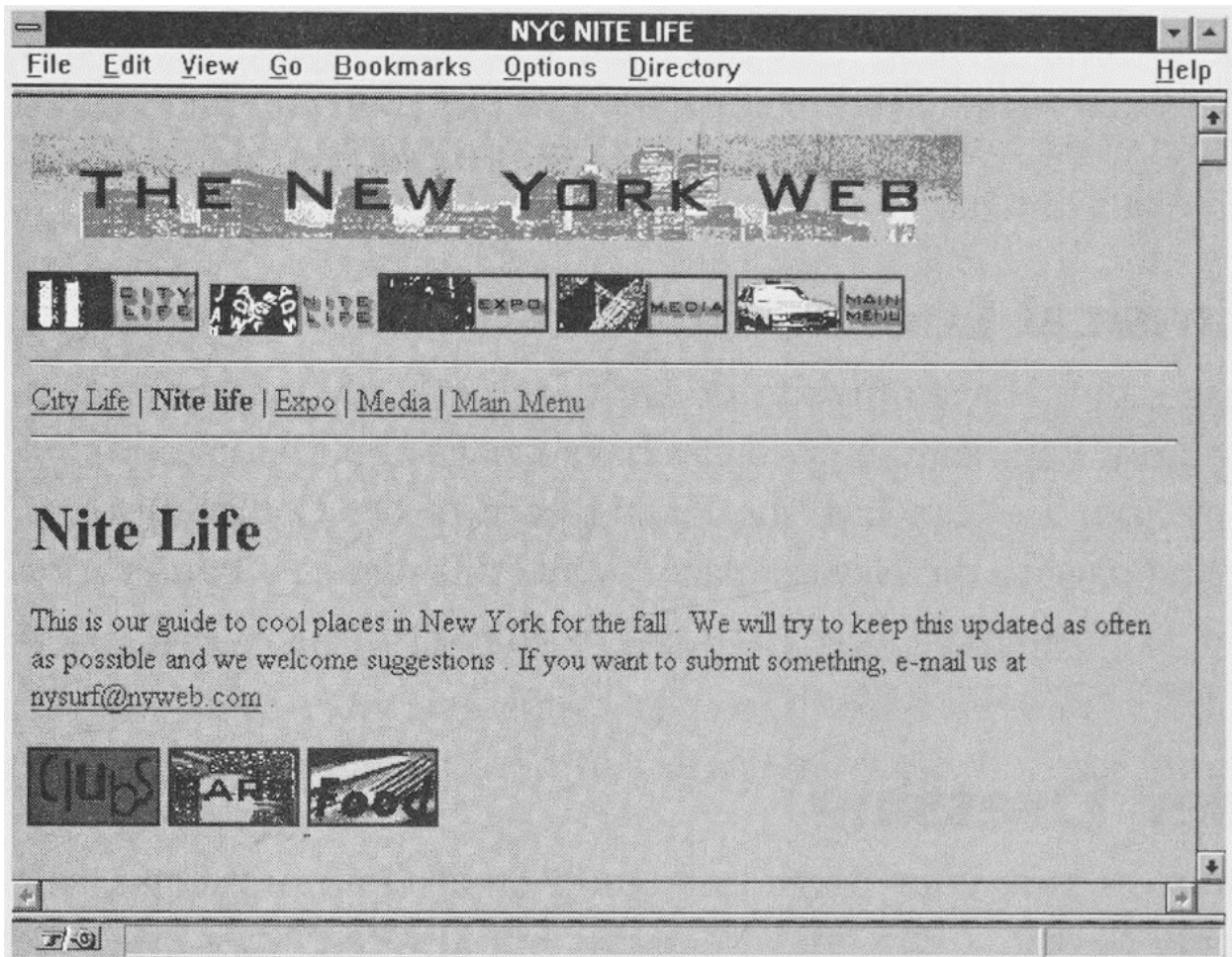
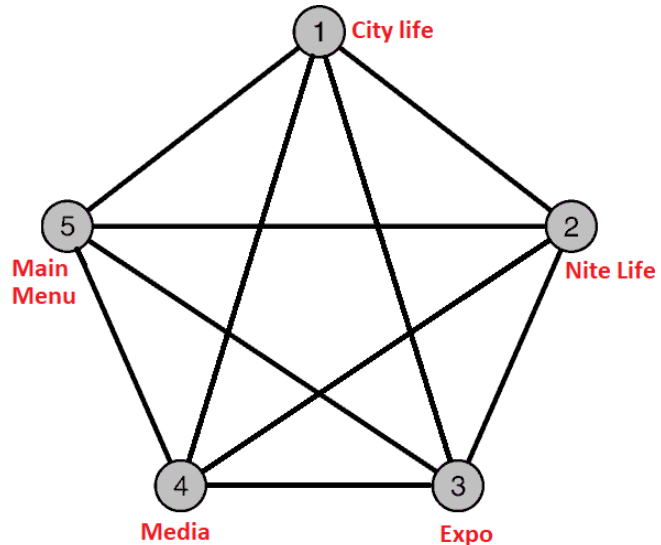


Figure 9-3 The New York Web, where each Web page contains links to other major sections

(Fox, p.239, Fig. 9-3.) Figure 9-3 shows an example in which “[a] set of custom icons at the top of the screen leads to the City Life, Nite Life, Expo, Media, or Main Menu section. This same menu is printed at the top of every page.” (Fox, p.238.) These links are also shown under the icons as textual hyperlinks (as indicated by the underlining). In the example of Figure 9-3, the user can reach any one of the five

web pages in the set (*i.e.*, City Life, Nite Life, Expo, Media, or Main Menu) from any other page. In the example above, the “Nite Life” page contains links to the other four pages, and the same holds true for each of those other pages. Each page therefore contains links to each other page (as shown in the figure at right that I created for illustration).



176. It therefore would have been obvious to implement the proposed combination such that the “**show structure of nodes**” as described for claim 1[a] also “**includes a concurrent path of nodes.**” During operation of the system based on the proposed combination, therefore, the nodes would have been navigated and displayed either (a) automatically using the default sequential path specified by the node and link information in Shimizu, as explained for claim 20 above, or (b) by utilizing hyperlinks within the nodes to skip to one or more other nodes, thus taking advantage of one or more concurrent node paths.

177. I provided a full explanation of the motivation to combine Shimizu, Benedict and Fox for claim 1[a], which applies here. An ordinarily skilled artisan

would have been further motivated to implement the proposed combination to provide a “**concurrent path of nodes**” to enhance flexibility and the user’s experience. The ability to follow multiple different paths to navigate through a set of information was one of the key features that motivated the creation of hypertext systems. Benedict provides an express motivation to provide this functionality by explaining that HyperShow links “enable you to control the flow of your presentation spontaneously, without editing the original slide order.” (Benedict, pp.435-36.) Benedict touts “[t]he capability of viewing slides in any sequence” as a clear benefit because it “turns an order-driven presentation into a presentation driven by information,” and lets the user “tailor the information to accommodate the interests and focus of your audience.” (*Id.*)

178. Fox provides a similar express motivation by describing consistent hyperlinked navigation tools on each page, such as the example in Figure 9-3 above, as a way to achieve consistency and improve user experience. (Fox, pp.238-239.) An ordinarily skilled artisan would have appreciated that providing for one or more concurrent paths through the nodes have provided a useful enhancement to the proposed combination which otherwise would be limited to the sequential path established by the node and link information. The ability to modify the path through the nodes would have provided useful adaptability, for example, by allowing the

system to skip past nodes that the user finds less relevant or interesting, thus allowing a more focused and useful presentation of content for the user.

C. Ground 2: Claims 22-29, 32-45 Are Obvious Over Shimizu, Fox, Benedict, Wolfe and Guinan

1. Independent Claim 22

179. For reference, claim 22 recites:

22. A method for providing in a sequential arrangement to a user's computer, content derived via a communications network from a plurality of resources, the method comprising the steps of:

- [a] receiving a request for a user defined online search from a remote user;
- [b] providing a plurality of search results and their corresponding resource locators in response to said request, automatically creating a show structure of nodes based on said search results, each node identifying a resource from a plurality of accessible resources via said communications network;
- [c] without requiring user input automatically accessing a plurality of said resources;
- [d] retrieving content corresponding to each of said accessed resources; and
- [e] automatically accessing and presenting said retrieved content through said user's computer in accordance with said show structure such that

during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.

(’932, 18:54-19:9 (Claim 22; bracketed notation (e.g., [a]) added).) Many of the limitations of claim 22 are substantially similar to limitations recited in claim 1 that were covered above. In the interests of brevity and consistency, therefore, I will refer back to my analysis of claim 1 where appropriate.

180. The preamble of claim 22 recites “[a] **method for providing in a sequential arrangement to a user’s computer, content derived via a communications network from a plurality of resources....**” As explained for the preamble of claim 1, the proposed combination renders obvious a method of providing to a user’s computer content derived from a plurality of Web resources (e.g., web pages, images, etc.) over the Internet (“**via a communications network**”). As also explained, the proposed combination renders obvious displaying content “in a sequential arrangement” because, as noted above, node content is displayed in a user in a particular sequence as indicated by the link information. (Shimizu, *e.g.*, 7:62-65, 9:1-4; *see also* Benedict, p.19 (“ScreenShows are on-screen presentations in which the program displays each slide in order.”).)

- (a) “receiving a request for a user defined online search from a remote user;” (Claim 22[a])
- (b) “providing a plurality of search results and their corresponding resource locators in response to said request...” (Claim 22[b][1])

181. These two limitations are closely-related and thus will be addressed together. I am informed that a district court in prior litigation construed “**online search**” as “*Internet search*.”

182. Claim 22[a] and 22[b][1] would have been obvious in view of Fox and Guinan, which together confirm that receiving a request for an on-line search from a remote Internet user, and then providing a plurality of search results, was well-known to ordinarily skilled artisans. As Fox explains:

Search Indexes

Hypertext makes it easy to get lost. That’s part of the appeal. You can start out reading about Sperm Whales and end up reading about the Prince of Wales. But what if you want to find pages dealing with a specific topic?

Luckily, the Web is full of indexes. An index is a special Web page that allows you to type in a search term or keyword. The index then spits out a list of Web pages matching your search. You can then click on any page that catches your interest and transfer there automatically...

Many indexes allow you to search the full contents of thousands of Web pages. Some indexes have a series of forms or menus you can

customized, allowing you to find exactly what you want.

(Fox, pp.10-11.) Fox identifies more than 10 existing Internet search indexes, including the Lycos and WebCrawler search engines that were popular by the mid-1990s. (Fox, pp.11-12, Table 1-1.) But Fox does not describe how results from an on-line search are used beyond being displayed by the user. I have therefore cited Fox in combination with Guinan, which discloses a technique for (a) allowing a user to enter a natural language search query and then (b) searching for hypertext nodes in response to the query to create a “guided tour” for the user. (Guinan, pp.124-125.)

183. Guinan explains “[i]n using any hypertext system an end-user or browser will encounter many technical problems that have been well-documented elsewhere,” such as “disorientation and the quality of the information retrieved by the user from the hypertext.” (Guinan, p.122, §1.) This problem with hypertext was well-known and, as acknowledged in Fox, one of the key reasons the industry adopted on-line searching in the Web context as discussed above. (Fox, pp.10-11 (“Hypertext makes it easy to get lost... [W]hat if you want to find pages dealing with a specific topic? Luckily, the Web is full of indexes.”).)

184. Guinan states that, to address problems of hypertext systems:

We propose a method for dynamically generating a guided tour through

a hypertext in response to a user's query which will plan a tour through a hypertext covering all the nodes the system believes relevant to the user's information need as expressed in a natural language query, which eases disorientation problems by using the guided tour metaphor... and which takes advantage of typing of information links in the hypertext to present the nodes on the tour in a logical fashion. Effectively we extend the guided tour metaphor by including the information retrieval features... and we add in logical sequencing of nodes on the tour to complete the process.

(Guinan, p.123, §1 (Introduction).)

185. The first step of Guinan's proposed method is for the user to enter a "natural language" search query. (Guinan, p.124, §3.) The hypertext nodes are then searched for keywords matching the query, which are scored using a weighting algorithm. (*Id.*; *see also id.*, p.129, §6 ("*Finding information*: When the user enters a search query in the form of a natural language statement it is matched to the nodes in the hypertext using standard information retrieval techniques.") (italics in original).) Guinan explains that, in weighting hypertext nodes, search terms that appear less frequently throughout the collection are given more weight using known

TF-IDF techniques,⁶ and as such, hypertext nodes containing more of these less frequent terms are given higher weights. (*Id.*) Hypertext nodes are also weighted based on the number of connections they have to other nodes. (*Id.*) Nodes whose weights satisfy a threshold value are included in the guided tour. (Guinan, p.125 (“All nodes whose weight is greater than or equal to this cut-off or threshold value would be selected for inclusion in the tour.”).)

186. Once matching hypertext nodes have been identified, a guided tour is assembled. “At this stage, the nodes for the guided tour have been dynamically chosen in response to a user’s query and some form of logical ordering has to be imposed on the selected tour nodes.” (Guinan, p.125, §3.) Guinan discloses an ordering procedure that takes into account linking relationships between the matching nodes. (*Id.*) “This procedure is followed until every node selected for inclusion in the tour has been positioned correctly relative to each other node in the tour.” (*Id.*) Guinan thus discloses receiving a request for a search, and then using

⁶ Guinan uses a weighting technique referred to as TF-IDF, an acronym for term frequency-inverse document frequency. (Guinan, p.124.) TF-IDF is a well-known technique for assessing the importance of terms appearing in documents. The basic idea behind TF-IDF is that terms appearing less frequently in documents are more likely to be important to the documents than terms that appear more frequently. TF-IDF will thus assign higher weights to these infrequently appearing terms.

that query to identify hypertext nodes for creating a guided tour for presenting those nodes.

187. *Rationale and Motivation to Combine (Shimizu, Benedict, Fox, and Wolfe with Guinan)*: I provided a full analysis of the manner and motivation to combine with respect to Shimizu, Benedict, Fox, and Filepp in my discussion of claim 1 above, and that analysis applies here. With respect to claim 22[a] and 22[b][1], it would have been obvious to combine these references with Guinan, predictably resulting in the system based on the proposed combination adapted to receive a request for a user-defined online search from a remote user, and then provide a plurality of search results and their corresponding resource identifiers in response.

188. Guinan does not expressly disclose that the hypertext nodes in the guided tour are located using an “**online search**” to the extent this term is interpreted as requiring an Internet search. But these details would have been trivial and obvious in view of Fox which, as mentioned above, expressly discloses online search indexes accessible over the Web, which provided results in response to search requests from remote users over the Internet. As noted below in my discussion of the motivations to combine, moreover, Guinan was expressly designed to solve problems with hypertext systems – and the Web is a hypertext system. It would also have been

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

obvious that the search results obtained by Guinan, under the combination with Fox, would have been accompanied by the URLs (“**resource identifiers**”). This is because, as was elementary knowledge to ordinarily skilled artisans, resources on the Web are identified using URLs. (Fox, p. 13 (“A *URL* (or *Universal Resource Locator*) is the address of a document, file, or other Internet resource... The World Wide Web gives every article, newsgroup, database, computer, file, or other Internet fragment its very own URL.”).) It thus would have been obvious to adapt the hypertext searching and guided tour techniques of Guinan, predictably resulting in an “**online search**” that receives hypertext nodes as search results as well as their corresponding “**resource identifiers**,” *i.e.*, URLs.

189. Guinan is an analogous reference in the same field of providing organized or sequenced presentation of hypertext content, and would have been reasonably pertinent to a number of problems facing the inventor. For example, Guinan would have been reasonably pertinent to the problem of how to use searching techniques to create a list of hypertext nodes for creating a show structure.

190. The motivation to combine would have been straightforward. Guinan provides an express motivation to combine by explaining that, using her techniques for creating a guided tour based on a search, “[u]sers don’t need to worry about getting lost or confused in the hypertext as all they need to do is follow the tour

which is generated in response to their query.” (Guinan, p.129, §6.)

191. This motivation is also expressly echoed in Fox, which also identifies the use of searching techniques to help users locate specific information and avoid getting lost. (Fox, pp.10-11 (“Hypertext makes it easy to get lost... [W]hat if you want to find pages dealing with a specific topic? Luckily, the Web is full of indexes.”).) The fact that Guinan and Fox both relate to hypertext systems (the Web being a hypertext system as discussed above), and that both address a common problem and identify search as the solution, reinforces their combinability and provides a further motivation to combine.

192. An ordinarily skilled artisan would thus have been motivated to use the searching techniques in Guinan and Fox to more easily identify relevant hypertext nodes for the user. These nodes, as explained below, are in turn used to create the claimed “**show structure of nodes**” that is delivered and displayed to the user under the proposed combination.

193. An ordinarily skilled artisan would also have been motivated to adapt the teachings of Guinan to an Internet search index that identifies search results with URLs, as disclosed in Fox. An ordinarily skilled artisan would have found this to have been a natural adaptation considering that, as Fox confirms, the Web is an example of a hypertext-based system. (Fox, p.9 (“The World Wide Web uses

hypertext to link all of its pages together.”).) An ordinarily skilled artisan would have been motivated to adapt Guinan for the Web to take advantage of the commercial growth and growing popularity of the Web, as fully explained for claim 1, and to expand the audience for Guinan’s guided tour and the universe of content that could have been incorporated into the tour.

194. An ordinarily skilled artisan would have had at least a reasonable expectation of success and could have implemented the proposed combination using conventional programming techniques. The ability to search through hypertext documents was well-known, as evidenced by Guinan and the many commercially-available examples of Internet search indexes in Fox. Keyword searching techniques were well-known decades before the priority date, with the TF-IDF term weighting techniques in Guinan dating back to at least the 1970s. And as explained for claim 1, the use of URLs to identify resources on the Web was also well-known. The disclosures of Fox and Guinan would have provided more than sufficient details to have enabled an ordinarily skilled artisan to implement the details I have cited for this claim limitation, without undue experimentation.

- (c) **“...automatically creating a show structure of nodes based on said search results, each node identifying a resource from a plurality of accessible resources via said communications network” (Claim 22[b][2])**

195. The second part of claim 22[b] is substantially similar to claim 1[a] (**“creating a show structure of nodes, each node identifying a resource from a plurality of accessible resources, at least some of which are network accessible resources”**). As explained for claim 1[a], the proposed combination discloses **“creating a show structure of nodes... each node identifying a corresponding resource via said communications network,”** because it discloses creating a set of nodes that define their associated content and their links to other nodes. As fully discussed above, the node and link information created in Shimizu identifies a set of nodes, the content associated with each node, and the links that specify the order in which the nodes are displayed. It would also have been obvious based on the further combination with Fox, as explained, that the resource could have been a web page, image and other resource identified by a URL and accessible over the Internet (**“said communications network”**). With respect to the act of **“creating”** the claimed show structure, it would have been obvious to an ordinarily skilled artisan that the show structure in Shimizu was **“creat[ed],”** as it must have been brought into existence in order to perform the node display and other functions described in

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

Shimizu. As explained below, the show structure would have been created automatically under the further combination with Guinan.

196. It would also have been obvious to create the show structure “**automatically**” and “**based on said search results.**” As explained for claim 22[a] and 22[b][1], Guinan discloses automatically constructing a guided tour of hypertext nodes in response to a search query. It would also have been obvious that this occurs “**automatically**” because Guinan requires no user action to build the guided tour based on the search results. It therefore would have been obvious that to provide the guided tour created by Guinan, which identifies the hypertext nodes and specifies the order in which they will be presented, to the system of Shimizu to use as node and link information for the claimed “**show structure.**” Although the user enters a search query, this is a separate and unclaimed act that precedes the claimed step of creating a show structure. I note that the ’932 patent similarly contemplates user action in initiating searches used to create show structures. (’932, 11:63-66 (“In accordance with one embodiment of the invention, the RoboGuide program may prepare various show structures based on a user’s search specifications.”).) Because the hypertext nodes and their links were identified via the guided tour in Guinan, under the proposed combination, the user would not need to take any action beyond entry of the initial search query. Under the proposed combination of Guinan with

Shimizu, Benedict and Fox, the results obtained from the online search would have been used to identify nodes and automatically create the “**show structure of nodes.**” The rationale and motivation to combine as to Guinan was provided for claim 22[a] and 22[b][1] above. A person of ordinary skill in the art would have been motivated to adapt the on-line feature, as explained above, because it provides a useful technique for identifying and sequencing the nodes for the claimed show structure. (Guinan, p.129, §6.) As a further motivation to combine, I note that the “guided tour” of hypertext nodes in Guinan is directly analogous to the show structure of nodes under the proposed combination. It therefore would have been obvious to use the search results provided by Guinan – which Guinan placed into an ordered and sequenced guided tour – as the basis for automatically creating the claimed show structure of nodes under the proposed combination.

- (d) **“without requiring user input automatically accessing a plurality of said resources;” (Claim 22[c])**
- (e) **“retrieving content corresponding to each of said accessed resources; and” (Claim 22[d])**
- (f) **“automatically accessing and presenting said retrieved content through said user’s computer in accordance with said show structure....” (Claim 22[e][1])**

197. These limitations are substantially similar to limitations already covered in claim 1, discussed in detail above, and the analysis above applies equally

here. As explained, under the proposed combination, a plurality of resources would have been automatically accessed in order to access and retrieve the content corresponding to those resources, in order to present that content to the user's computer. The content, as explained, is presented through the user's computer in continuous sequential fashion, one node after another, without requiring any input from the user beyond an initial click of a PLAY button or other control to commence the node display process. It would have also have been obvious that, in order to display content as disclosed in Shimizu, Benedict, and Fox, that content would have been accessed and retrieved. The resources would also have been delivered **"in accordance with said show structure,"** because the resources are identified by the node information and are presented to the user's computer in the sequence specified by the corresponding link information.

(g) **"...such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed." (Claim 22[e][2])**

198. This final aspect of claim 22[e] is identical to language at the end of claim 1[c] described above. As explained, this limitation is disclosed and obvious based on the disclosures in Benedict and Wolfe.

2. Claim 23

199. Claim 23 depends from claim 22, and further recites “[t]he method in accordance with claim 22, further comprising the step of causing said content to be presented through said user’s computer.” As explained for claim 22[c]-[e] above, the content is presented through the user’s computer.

3. Claims 24-25

200. Claim 24 depends from claim 23, and further recites “wherein said step of presenting said content further comprises the step of providing duration information, representing the duration that corresponding content to said resource is presented.” Claim 25 depends from claim 24, and further recites “wherein said step of providing duration information provides an interactively variable duration information, representing the duration that corresponding content to said resource is presented so as to enable a user to vary said duration.” These limitations were fully covered in my discussion of claim 1[d]. As explained, the proposed combination renders obvious providing “duration information,” *e.g.*, a display time that content associated with a node will be presented before advancing to the next node. This duration is also “interactively variable duration information” because the user can change it.

4. Claim 26

201. Claim 26 is substantially similar to claim 2 as shown below (overlapping language shown in underlining):

Claim 26	Claim 2
<u>The method in accordance with claim 25, wherein said step of providing an interactively variable duration information further comprises providing a duration information for each of said accessed resources, representing the duration that corresponding content to said resource is presented.</u>	<u>The method in accordance with claim 1, wherein said step of providing interactively variable duration information further comprises the step of providing interactively variable duration information for each of said accessed resources, representing the duration within which a corresponding content to said resource is presented.</u>

202. As shown, claim 26 is substantially similar to claim 2, with minor differences that are not material to my analysis. The limitations of claim 26 are thus disclosed and rendered obvious for the reasons provided for claim 2.

5. Claim 27

203. Claim 27 depends from claim 22, and further recites “[t]he **method in accordance with claim 22, wherein said step of accessing said resources further comprises the step of downloading a file from a server residing within said communications network.**” Claim 27 adds nothing of patentable significance as it merely describes well-known techniques for retrieving content resources from the Web. Under the further combination with Fox, the step of accessing the resources

would have included downloading a file from a server residing on the Internet identified in the URL. (Fox, p.14 (“The second part of most URLs is the full Internet address of the machine where the file is located.”).)

204. It would also have been obvious that these resources would have taken the form of “a **file**,” because files are fundamental units of data storage in any computer system, and are used by web servers and browsers to store resources for the Web. (Fox, pp.46-47 (identifying standard “file formats” for images, text, movies, etc.), p.118 (describing how to view an HTML web page in a web browser and “save the entire HTML file to your hard disk”); *see also Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.164 (“**file** A complete, named collection of information, such as a program, a set of data used by a program, or a user-created document. A file is the basic unit of storage that enables a computer to distinguish one set of information from another.”).)

6. Claim 28

205. Claim 28 depends from claim 22, and further recites “[t]**he method in accordance with claim 22, further comprising the step of transmitting to said user dynamic content.**” This feature is rendered obvious based on the disclosures in Fox, which confirms that it was well-known that web servers on the Internet generated content in response to requests from web browsers, such as a request to

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

access a resource at a particular URL. Fox describes a process using a well-known process known as the Common Gateway Interface (CGI), which allowed web servers to execute scripts that dynamically generated web content on-the-fly in response to a request from the web browser. (Fox, p.391, p.423 (“CGI scripts are used for doing all of the ‘cool’ stuff on the Net. There are... sites that create graphic images on-the-fly.”).) For example, Fox shows an example of a web page that uses a CGI script to generate a web page image representing the current image captured by a camera in a laboratory. (Fox, p.424, Fig. 15-2.) Aside from images, it was well-known (and in fact more common) to use CGI techniques to generate HTML on-the-fly in response to a user request, which is discussed extensively in Fox. (Fox, pp.421-23.) It would have been obvious to generate new content for a wide variety of purposes, such as news, financial, and weather content, as these are areas where information regularly changes.

206. It this would have been obvious that the proposed combination would have transmitted “**user dynamic content**,” *e.g.*, content generated on-the-fly for the requesting user. An ordinarily skilled artisan would have been motivated to provide “**dynamic**” content because dynamic content would have been more likely catch the interest of the user. Fox itself provides multiple express motivations to employ this technique, as it explains that “CGI scripts are used for doing all of the ‘cool’ stuff

on the Net,” such as creating graphic images on-the-fly, accessing live video feeds, accessing large databases, among other uses. (Fox, p.423.) The use of CGI to generate dynamic content on-the-fly for users was well-known, and an ordinarily skilled artisan would have been motivated to use this technique to provide dynamic content to the requesting user – a clear advantage when compared to simply sending the same (static) content to every requesting user.

7. Claim 29

207. Claim 29 depends from claim 28, and further recites “**the step of inserting resource identifiers interspersed within said show structure relating to said dynamic content.**” Claim 29 adds nothing of patentable significance as it merely identifies known well-known functionality of using URLs (“resource identifiers”) to identify Web resources.

208. Fox explains that “[a] *URL (or Universal Resource Locator)* is the address of a document, file, or other Internet resource... The World Wide Web gives every article, newsgroup, database, computer, file, or other Internet fragment its very own URL.” (Fox, p.13 (*italics in original*).) To include a graphic image on a web page, for example, the HTML of the web page may include a URL identifying the file for the image. (Fox, pp.326-327.) Fox provides the following exemplary HTML:

(Fox, p.327.) This exemplary HTML tells the web browser to retrieve the graphical image file identified by the URL, in this case, a file called “Jupiter.gif” from the “astronomy/planets” directory at the “www.smartpants.edu” server on the Web. The URL in this case provides an example of a **“resource identifier.”**

209. Interspersing of URLs to identify Web resources was basic knowledge of ordinarily skilled artisans. Under the proposed combination, therefore, it would have been obvious to **“insert[] resource identifiers interspersed within said show structure relating to said dynamic content.”** As explained for claim 1[a], the show structure corresponds to a set of nodes and their links or connections to other nodes, based on the combination of Shimizu, Benedict and Fox. And in the context of Web resources, as noted, these nodes would have identified Web resources (*e.g.*, pages, images, etc.) via URLs. It therefore would have been obvious to insert (**“intersperse”**) URLs within the show structure to identify the dynamic content. (Fox, p.391, p.423 (“CGI scripts are used for doing all of the ‘cool’ stuff on the Net. There are... sites that create graphic images on-the-fly.”).) As explained for claim 28 above, Fox provides an example of a CGI script used to generate a dynamic image for a web page representing the current image captured by a camera. (Fox, p.424, Fig. 15-2.) The URLs for these resources (**“resource identifiers”**) would have been **“interspersed within said show structure”** because, under the proposed

combination, they would have been stored within the show structure, *e.g.*, via the entity information pointer slot **63** of Shimizu as fully described for claim 1[a].

8. Claim 32

210. Claim 32 depends from claim 22, and further recites “**the step of storing said content for each resource accessed.**” This limitation adds nothing of significance because it would have been obvious that the content for each resource accessed would have been stored. I note that claim 32 does not specify any particular location where storage must take place. Under the proposed combination of Shimizu, Benedict, and Fox, it would have been obvious that the content for each accessed resource would have been stored in multiple places including at least (1) the server on the Internet that must store the content for each resource in order to make that content available to the user computer for download, and (2) the user computer that downloads the content from the server in order to access it and display it to the user. In both cases, it would have been obvious that the proposed combination would have performed the step of storing the content, either temporarily in volatile memory (such as RAM) or in non-volatile storage (such as a hard disk or other persistent storage), by both the server and the user computer.

211. With respect to storage on the user device, Fox further confirms that storage of content for accessed resources was a built-in feature of conventional web

browser software, which were capable of storing all web content downloaded by and displayed by the web browser including web pages and graphic images (“**each resource accessed**”). This would have been known by even end-users familiar with the history or web cache features of popular web browsing programs. For example, with respect to one conventional web browser, Fox explains: “Once you retrieve a Web page, SkipKnot saves it on your hard disk—graphics and all—for as long as you want to keep it.” (Fox, p.82; *see also id.*, p.93 (describing a History List feature showing previously-visited and locally stored web pages).) With respect to the then-popular Netscape web browser, Fox explains that “[w]henver you retrieve a Web page, images are stored in memory. If you return to a page you’ve already seen, images load much faster due to this *memory cache*.” (Fox, p.138 (italics in original); *see also id.* p.164 (“Memory cache [for Mosaic]: Speed up page reloading by storing each page that is read.”).) Under the proposed combination, therefore, it would have been obvious to store a data file for each resource accessed (*e.g.*, web page, image, etc.) as part of the operation of the web browser.

9. Claims 33-35

212. Claims 33-35 are substantially similar to claims 7-9 as shown below (overlapping language shown in underlining):

Claim 33	Claim 7
<u>The method in accordance with claim 22, wherein one or more of said nodes identifies video content.</u>	<u>The method in accordance with claim 1, wherein one or more of said nodes identifies video content.</u>
Claims 34	Claim 8
<u>The method in accordance with claim 22, wherein one or more of said nodes identifies audio [content].⁷</u>	<u>The method in accordance with claim 1, wherein one or more of said nodes identifies audio content.</u>
Claim 35	Claim 9
<u>The method in accordance with claim 22, wherein one or more of said nodes identifies displayable content.</u>	<u>The method in accordance with claim 1, wherein one or more of said nodes identifies displayable content.</u>

213. As shown, claims 33-35 are identical to claims 7-9, respectively, other than their claim dependencies. The limitations of claims 33-35 are thus disclosed and rendered obvious for the reasons provided for claims 7-9.

10. Claims 36-37

214. Claims 36-37 is substantially similar to claims 17-18, respectively, as shown below (overlapping language shown in underlining):

⁷ Claim 34 of the '932 patent as originally issued concludes with “**identifies audio come,**” with “come” being corrected to “**content**” in a Certificate of Correction signed on June 9, 2015.

Claim 36	Claim 17
<u>The method in accordance with claim 22, wherein said computer is a mobile device.</u>	<u>The method in accordance with claim 1, wherein said computer is a mobile device.</u>
Claim 37	Claim 18
<u>The method in accordance with claim 22, wherein said content is presented through said user's computer using a web browser.</u>	<u>The method in accordance with claim 1, wherein said content is presented through said user's computer using a web browser.</u>

215. As shown, claims 36-37 are identical to claims 17-18, respectively, except for their claim dependencies. The limitations of claims 36-37 are thus obvious for the reasons provided for claims 17-18.

11. Claims 38-40

216. Claim 38 depends from claim 22, and further recites “**wherein said user request comprises a search of accessible Internet content.**” Claim 39 depends from claim 22, and further recites “**wherein said user request comprises a search of a web site.**” Claim 40 depends from claim 22, and further recites “**wherein said user request comprises a search of a remote server.**”

217. As explained for claim 22[a]-[b], under the proposed combination, the user request would have comprised a search of Web content accessible through the Internet (*e.g.* web pages, images, etc., accessible from one or more websites) located on one or more remote web servers on the Internet. (Fox, *e.g.*, pp.11-13.)

12. Claims 41-42

218. Claim 41 depends from claim 22, and further recites “[t]he method in accordance with claim 22, wherein said show structure is presented to said user.” Claim 41 is substantially similar to claim 10, which recites that “said show structure is displayed to the user.” Claim 41 is obvious for the same reasons.

219. Claim 42 depends from claim 41, and further recites “wherein said show structure is presented to said user while said content is being delivered.”

This is also shown through Figure 8 of Shimizu, which again shows:

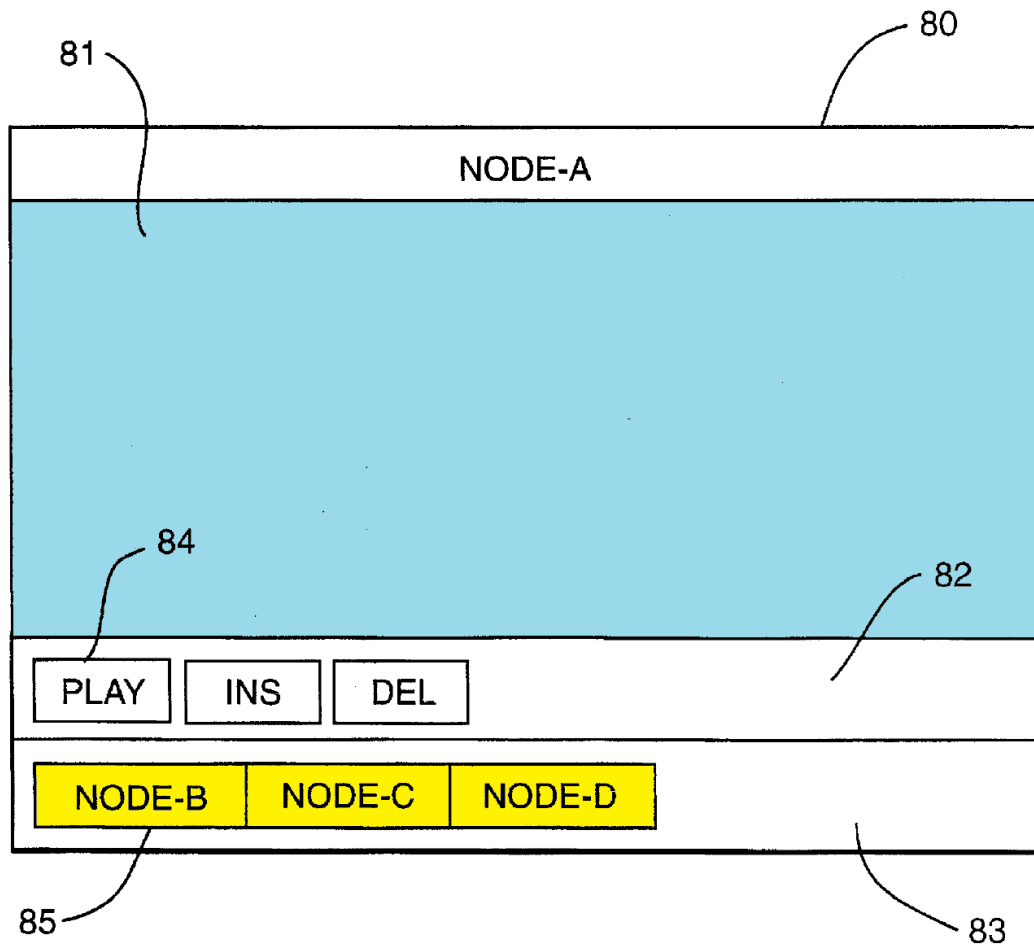


FIG. 8

(Shimizu, Fig. 8 (highlighting added).) As explained for claim 10, Figure 8 depicts the “**show structure of nodes**” because it displays icons **85** showing icons for the other nodes and the order in which they are linked to NODE-A. Figure 8 also depicts display area **81** for presenting the content of NODE-A, the currently displayed node. (Shimizu, 7:16-18 (“The window **80** is made up of node content display area **81**...”), 7:20-24, 7:62-65, 7:52-59 (“...the content of the acquired information is reproduced

and displayed in the node content display area **81** of the window **80**.”).) The proposed combination therefore renders obvious presenting the show structure to the user “**while said content is being delivered**,” *e.g.*, content of NODE-A.

13. Claims 43-44

220. Claims 43-44 are substantially similar to claims 20-21, respectively, as shown below (overlapping language shown in underlining):

Claim 43	Claim 20
<u>The method in accordance with claim 22, wherein said show structure includes a sequential path of nodes</u>	<u>The method in accordance with claim 1, wherein said show structure includes a sequential path of nodes.</u>
Claims 44	Claim 21
<u>The method in accordance with claim 22, wherein said show structure includes a concurrent path of nodes.</u>	<u>The method in accordance with claim 1, wherein said show structure includes a concurrent path of nodes.</u>

221. As shown, claims 43-44 are identical to claims 20-21, respectively, other than their claim dependencies. The limitations of claims 43-44 are thus disclosed and rendered obvious for the reasons provided for claims 20-21.

14. Claim 45

222. Claim 45 depends from claim 22, and further recites “[t]he method in accordance with claim 22, wherein said show structure of nodes is customized based on user specified information.”

223. As explained for claim 29, Shimizu and Benedict disclose several ways in which the show structure can be customized based on user specified information. With respect to Shimizu, it discloses a DEL button (Fig. 8) that the user can activate to delete a node. (Shimizu, Fig. 8, 7:25-28 (“...the DEL button is used to delete a node.”).) Shimizu also discloses an INS button (Fig. 8) that allows the user to insert a new node at a particular position. (Shimizu, Fig. 8, 7:26-27, 9:5-59.) And Shimizu discloses a feature allowing the user (using a computer mouse) to change the order of nodes. (Shimizu, 7:40-48.) Similarly, Benedict discloses a Slide Sorter feature that allows the user to delete, add, or reorder slides in a presentation. (Benedict, pp.251-252.) And as explained for claim 22 above, Benedict discloses a HyperShow feature in which users can create customized hyperlinks to jump to other slides within the presentation. (Benedict, pp.435-36.) Each one of these actions provides a way in which, under the proposed combination, **“said show structure of nodes is customized based on user specified information,”** because the user can provide information to the system about the order of nodes, the identity of nodes to delete, and hyperlinks connecting to other nodes as discussed. Any one of these user actions would have resulted in further customization of the show structure.

D. Ground 3: Claims 46-62 Are Obvious Over Shimizu, Fox, Benedict, Wolfe, Guinan and Miller

1. Independent Claim 46

224. Claim 46 is similar to claim 22, but also adds limitations relating to **“providing a predefined list of categories,”** and then providing a plurality of search results relating to a category amount those categories.

225. For reference, claim 46 recites:

46. A method for providing in a sequential arrangement to a user’s computer, content derived via a communications network from a plurality of resources, the method comprising the steps of:

- [a] providing a predefined list of categories;
- [b] providing a plurality of search results based on user selections and their corresponding URLs in response to a user request related to a category among said predefined list of categories;
- [c] automatically creating a show structure of nodes based on said search results, each node identifying a resource from a plurality of accessible resources via said communications network, wherein said show structure is provided in response to said user request;
- [d] without requiring user input automatically accessing a plurality of said resources; and
- [e] automatically accessing and presenting content corresponding through

said accessed resources to said user's computer in accordance with said show structure such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.

('932, 20:6-27 (Claim 46⁸; bracketed notation (e.g., [a]) added).) Many of the limitations of claim 46 are substantially similar to limitations recited in claims 1 and 22 that were covered above. In the interests of brevity and consistency, therefore, I will refer back to my analysis of claims 1 and 22 where appropriate.

226. The preamble of claim 46 recites “[a] method for providing in a sequential arrangement to a user's computer, content derived via a communications network from a plurality of resources....” It is unclear if the preamble of claim 46 recites a claim limitation, but even if it does, it is disclosed by the prior art. The preamble of claim 46 is identical to the preamble of claim 22, and as such, the proposed combination thus renders obvious the preamble.

(a) “providing a predefined list of categories;” (Claim 46[a])

227. As explained for claim 22[a], Fox describes various on-line search

⁸ A Certificate of Correction signed on June 9, 2015 corrects claim 46 to insert the words “**some of**” between the words “**at least**” and “**said.**”

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

facilities available by the mid-1990s, including Yahoo – “the mother of all Web indexes.” (Fox, p.178.) Fox explains that “Yahoo has hundreds of links to pages in all of the following categories,” and then lists 19 categories including Art, Business, Computers, Society and Culture, among many others. (Fox, p.178.) Figure 8-2 of Fox shows a screen capture from Yahoo showing a list of subcategories within the “Society and Culture” category, each category containing documents:

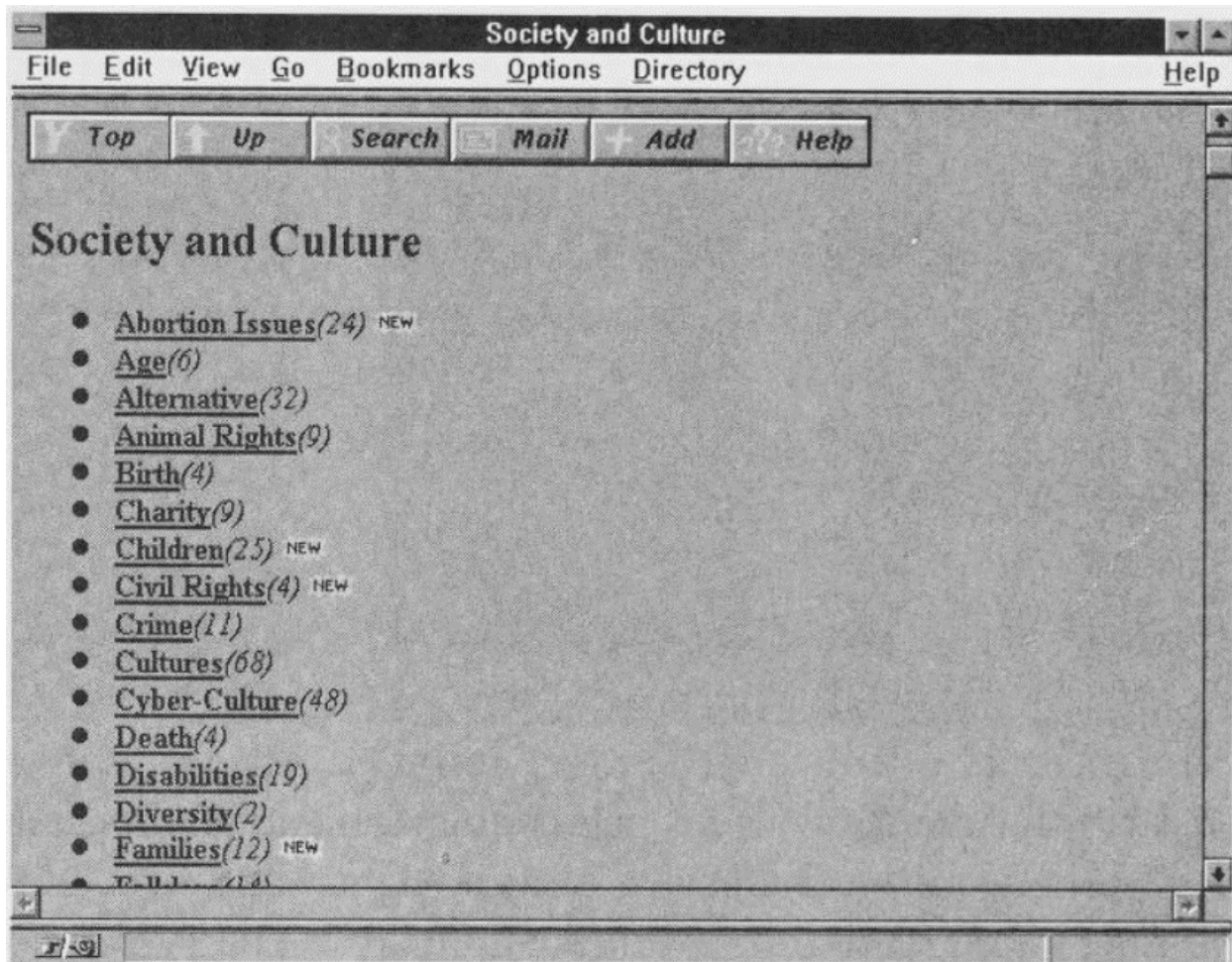


Figure 8-2 Yahoo

(Fox, p.179, Fig. 8-2.) But Fox does not disclose further details about Yahoo, such as the mechanism by which categories are selected for purposes of searching. I have therefore also cited to Miller for this limitation. As shown below, Miller discloses a technique for selecting categories (referred to in Miller as “sections”) within a predefined list of categories, which is then used to conduct a search.

228. As noted in **Part V.A.2** above, Miller describes various features of a commercial on-line service known as CompuServe which was popular in the mid-1990s. (Miller, p.1.) CompuServe provided a feature known as “forums,” which were on-line communities where members could exchange information and messages about particular topics of interest. (Miller, p.29 (“You can find forums on just about any topic, from scuba diving to C programming.”); *id.*, p.30 (“A forum is like one of those after-school clubs you belonged to back in high school. A bunch of people with the same interests get together to talk and exchange information.”).) Forums included a message section, “where forum members exchange messages with each other,” and a file library, “where forum files are stored.” (Miller, p.34.)

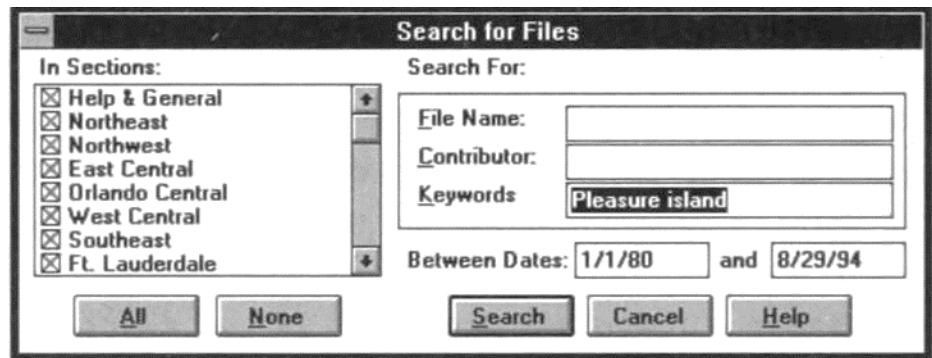
229. Miller discloses that, “[i]n addition, every forum’s message sections and file libraries are comprised of multiple *sections*. Each section represents a more focused area of interest within the forum topic.” (Miller, p.35 (*italics in original*).) “For example, the Florida Forum has sections for Ft. Lauderdale, Tampa, St.

Petersburg, and other specific locations.” (*Id.*) “The MacMillan Computer Publishing Forum (where you talk to Que [the publisher of Miller] has sections for Word Processing, Databases, Programming, and several other specific topics.” (*Id.*)

230. Miller also discloses features of CompuServe that allow users to search forums and sections within the forums. Figure 3.9 shows a user interface dialog box that allows a user to perform such a search:

Fig. 3.9

Use the Search for Files dialog box to search for specific forum files.



(Miller, p.42, Fig. 3.9; *see also id.*, p.42 (“A better way to find files... is to click the Search Libraries button in the toolbox. When you do this, a Search for Files dialog box appears (see fig. 3.9).”).) The dialog box, as shown, includes an “**In Sections:**” area that allows the user to select one or more sections to search. (Miller, p.42 (“Select the sections you wish to search.”).) “When you click the Search button, the forum’s libraries are searched for any and all files that match your search criteria.” (Miller, p.43; *see also id.*, p.419 (showing similar example).)

231. Miller thus discloses “**providing a predefined list of categories,**” *i.e.*,

a predefined list of sections (“**categories**”) from which the user could select as shown in Figure 3.9 above. A section in Miller qualifies as a “**category**” because it represents a class of items that have particular characteristics, *e.g.*, that relate to a particular topic of interest.

232. *Rationale and Motivation to Combine:* I already provided a full explanation of the rationale and motivation to combine Shimizu, Benedict, Fox, and Guinan in my discussion of claim 22 above. It would have been obvious to adapt in the system based on the proposed combination to provide the further ability to provide a list of predefined categories and allow the user to select from that list, as disclosed in Miller.

233. Miller is an analogous reference in the same field of techniques for organizing and presenting content, and would have been reasonably pertinent to a number of problems facing the inventor. For example, Miller would have been reasonably pertinent to the problem of how to implement a searching feature that allowed searches within selected categories.

234. An ordinarily skilled artisan would have been motivated to combine in the manner described above in order to improve the user experience. The ability to provide predefined categories for use in searching would have provided the clear benefit of allowing the user to focus a search on particular topics of interest, thus

resulting in search results that are more relevant to the user. This would have been particularly useful in cases where the search could result in a potentially large number of results. Using the example of Figure 3.9, suppose the user was only interested in files in the Florida Forum specifically pertaining to Ft. Lauderdale. The user could select just the Ft. Lauderdale section (and deselect other sections) to search only for files within that category, thus avoiding the need to review numerous search results from other sections not relevant to the user.

235. An ordinarily skilled artisan would have had at least a reasonable expectation of success. The ability to organize information into a predefined list of categories, and to search within selected categories, would have been considered rudimentary and elementary tasks to ordinarily skilled artisans. The disclosures of Miller would have provided more than sufficient details to have enabled an ordinarily skilled artisan to adapt the features that I have cited above to the proposed combination, without undue experimentation.

(b) “providing a plurality of search results based on user selections and their corresponding URLs in response to a user request related to a category among said predefined list of categories;” (Claim 46[b])

236. As explained for claim 22[b], based on the further combination with Guinan and Fox, it would have been obvious to provide a plurality of search results

with their corresponding resource locations (“URLs”). These search results, as explained for claim 22[b], would have been provided in response to a user request comprising a query provided by the user.

237. With respect to the further requirement that the search results be based on “**user selections**” and in response to “**a user request related to a category among said predefined list of categories,**” this would have been obvious in further view of Miller discussed above, which specifically discloses limiting the search to the categories selected by the user. Under the proposed combination, therefore, the search performed in response to a search query as disclosed in Guinan and Fox (as explained for claim 22[a]) would have been further enhanced to search for hypertext nodes within the selected categories. As explained, this would have provided a distinct advantage of focusing the search on items more relevant to the user.

- (c) “**automatically creating a show structure of nodes based on said search results, each node identifying a resource from a plurality of accessible resources via said communications network, wherein said show structure is provided in response to said user request;**”
(Claim 46[c])

238. The limitations in claim 46[c] are also recited in claim 22[b][2], except that claim 46[c] concludes by further reciting that the show structure “**is provided in response to said user request.**” This adds nothing of significance because, under

the proposed combination as explained for claim 22[b][2], the show structure would have been created and provided based on the search results, which in turn were obtained in response to the user's search query and category selections. Claim 46[c] is thus obvious for the same reasons as claim 22[b][2].

- (d) **“without requiring user input automatically accessing a plurality of said resources; and” (Claim 46[d])**
- (e) **“automatically accessing and presenting content corresponding through said accessed resources to said user's computer in accordance with said show structure such that during said presentation of at least said content from said accessed resources, at least some of said other network accessible resources are being accessed.” (Claim 46[e])**

239. These two final limitations of claim 46 are substantially similar to claim 22[c]-22[e], and are obvious for the same reasons provided above.

2. Claims 47-49

240. Claims 47-49 are substantially similar to claims 24-26, respectively, as shown below (overlapping language shown in underlining):

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

Claim 47	Claim 24
<u>The method in accordance with claim 46, further comprising the step of providing a duration information, representing the duration that corresponding content to said resource is presented.</u>	<u>The method in accordance with claim 23, wherein said step of presenting said content further comprises the step of providing duration information, representing the duration that corresponding content to said resource is presented.</u>
Claims 48	Claim 25
<u>The method in accordance with claim 47, wherein said step of providing a duration information provides an interactively variable duration information, representing the duration that corresponding content to said resource is presented so as to enable a user to vary said duration.</u>	<u>The method in accordance with claim 24, wherein said step of providing duration information provides an interactively variable duration information, representing the duration that corresponding content to said resource is presented so as to enable a user to vary said duration.</u>
Claim 49	Claim 26
<u>The method in accordance with claim 48, wherein said step of providing an interactively variable duration information further comprises providing a duration information for each of said accessed resources, representing the duration within which a corresponding content to said resource is being displayed.</u>	<u>The method in accordance with claim 25, wherein said step of providing an interactively variable duration information further comprises providing a duration information for each of said accessed resources, representing the duration that corresponding content to said resource is presented.</u>

241. As shown, claims 47-49 are substantially similar to claims 24-26, respectively, with the differences attributable to their claim dependencies. With

respect to claim 49, it concludes by reciting the duration “**within which**” the “**resource is being displayed**” whereas claim 26 concludes by reciting the duration “**that**” the “**resource is being presented**,” which present no material differences for purposes of my analysis. The limitations of claim 47-49 are disclosed and rendered obvious for the reasons provided for claims 24-26.

3. Claims 50-51

242. Claims 50-53 are substantially similar to claims 28-29, respectively, as shown below (overlapping language shown in underlining):

Claim 50	Claim 28
<u>The method in accordance with claim 46, further comprising the step of transmitting to said user dynamic content.</u>	<u>The method in accordance with claim 22, further comprising the step of transmitting to said user dynamic content.</u>
Claim 51	Claim 29
<u>The method in accordance with claim 50, further comprising the step of inserting resource identifiers interspersed within said show structure relating to said dynamic content.</u>	<u>The method in accordance with claim 28, further comprising the step of inserting resource identifiers interspersed within said show structure relating to said dynamic content.</u>

243. As shown, claims 50-51 are identical to claims 28-29, respectively, other than their claim dependencies. The limitations of claims 50-51 are thus disclosed and rendered obvious for the reasons provided for claims 28-29.

4. Claims 54-56

244. Claims 54-56 are substantially similar to claims 7-9 as shown below (overlapping language shown in underlining):

Claim 54	Claim 7
<u>The method in accordance with claim 46, wherein one or more of said nodes identifies video content.</u>	<u>The method in accordance with claim 1, wherein one or more of said nodes identifies video content.</u>
Claims 55	Claim 8
<u>The method in accordance with claim 46, wherein one or more of said nodes identifies audio content.</u>	<u>The method in accordance with claim 1, wherein one or more of said nodes identifies audio content.</u>
Claim 56	Claim 9
<u>The method in accordance with claim 46, wherein one or more of said nodes identifies displayable content.</u>	<u>The method in accordance with claim 1, wherein one or more of said nodes identifies displayable content.</u>

245. As shown, claims 54-56 are identical to claims 7-9, respectively, other than their claim dependencies. They are also substantially identical to claims 33-35, also addressed above. The limitations of claims 54-56 are thus disclosed and rendered obvious for the reasons provided for claims 7-9.

5. Claims 57-58

246. Claims 57-58 are similar to claims 17-18, respectively, as shown below (overlapping language shown in underlining):

Claim 57	Claim 17
<u>The method in accordance with claim 46, wherein said computer is a mobile device.</u>	<u>The method in accordance with claim 1, wherein said computer is a mobile device.</u>
Claim 58	Claim 18
<u>The method in accordance with claim 46, wherein said content is presented through said user's computer using a web browser.</u>	<u>The method in accordance with claim 1, wherein said content is presented through said user's computer using a web browser.</u>

247. As shown, claims 57-58 are identical to claims 17-18 except for their claim dependencies. They are also substantially identical to claims 36-37, also addressed above. The limitations of claims 57-58 are thus obvious for the reasons provided for claims 17-18.

6. Claims 59-61

248. Claims 59-61 are similar to claims 38-40, respectively, as shown below (overlapping language shown in underlining):

Claim 59	Claim 38
<u>The method in accordance with claim 46, wherein said user request comprises a search of accessible Internet content.</u>	<u>The method in accordance with claim 22, wherein said user request comprises a search of accessible Internet content.</u>
Claim 60	Claim 39

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

<u>The method in accordance with claim 46, wherein said user request comprises a search of a web site.</u>	<u>The method in accordance with claim 22, wherein said user request comprises a search of a web site.</u>
Claim 61	Claim 40
<u>The method in accordance with claim 46, wherein said user request comprises a search of a remote server.</u>	<u>The method in accordance with claim 22, wherein said user request comprises a search of a remote server.</u>

249. As shown, claims 59-61 are identical to claims 38-40 except for their claim dependencies. The limitations of claims 59-61 are thus obvious for the reasons provided for claims 38-40.

7. Claim 62

250. Claim 62 is similar to claim 10, as shown below (overlapping language shown in underlining):

Claim 62	Claim 10
<u>The method in accordance with claim 46, wherein said show structure is displayed to said user.</u>	<u>The method in accordance with claim 1, wherein said show structure is displayed to the user.</u>

251. As shown, claim 62 is identical to claim 10 except for their claim dependencies. Claim 62 is also substantially identical to claim 41, also addressed above. The limitations of claim 62 are thus obvious for the reasons provided for claim 10.

E. Ground 4: Claims 63-67 Are Obvious Over Shimizu, Fox, Benedict, Wolfe, and Sagman

1. Independent Claim 63

252. For reference, claim 63 recites:

63. A method for presenting in an ordered arrangement on a user's display, content derived via a communications network from a plurality of resources, the method comprising the steps of:

- [a] allowing the selection of at least one resource from a plurality of accessible resources via said communications network, wherein each of said resources is associated with a corresponding identifier displayed on a first section of said user's display;
- [b] providing an interface for creating a show structure of nodes each node identifying a corresponding resource, by allowing generation of a list of said resources by dragging and dropping said identifiers on a second section of said user's display;
- [c] without requiring user input, automatically accessing each of said resources identified by each of said nodes from said network accessible resources in accordance with said show structure;
- [d] automatically retrieving content corresponding to each of said accessed resources; and
- [e] causing the presentation of said retrieved content automatically in accordance with said show structure such that during said presentation

of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.

(’932, 21:5-22:7 (Claim 63⁹; bracketed notation (e.g., [a]) added).)

253. The preamble of claim 63 recites “[a] **method for presenting in an ordered arrangement on a user’s display, content derived via a communications network from a plurality of resources...**” Many of the limitations of claim 63 are substantially similar to limitations recited in claims 1 and 22 that were covered above. In the interests of brevity and consistency, therefore, I will refer back to my analysis of earlier claims where appropriate.

254. As fully explained for the preamble of claim 1, the proposed combination of Shimizu, Benedict and Fox renders obvious a method for “**presenting in an ordered arrangement on a user’s display, content,**” *i.e.*, content such as text, pictures, motion pictures, etc. The content is displayed “**in an ordered arrangement**” because, under the proposed combination as fully explained for claim 1, the content is presented in connection with the display of nodes presented in a specified order.

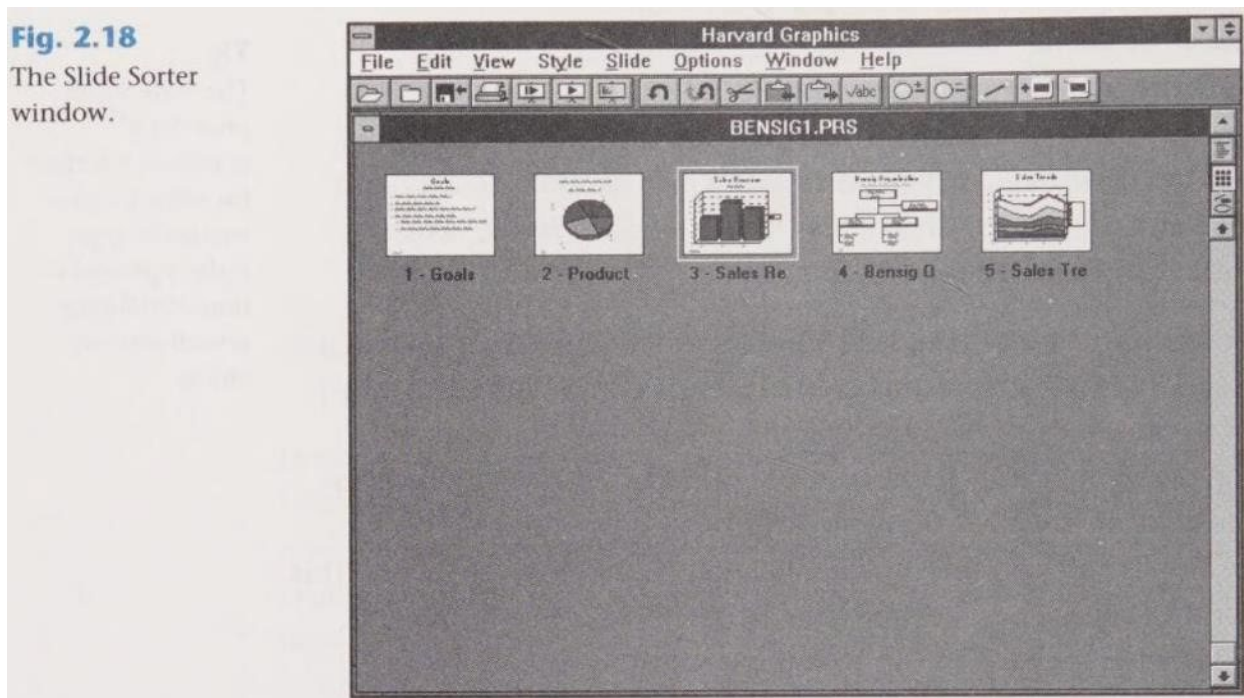
⁹ A Certificate of Correction signed on June 9, 2015 corrects claim 63 to insert the word “**in**” between the words “**automatically**” and “**accordance.**”

255. The content is also “**derived via a communications network from a plurality of resources,**” as recited. As explained, under the proposed combination with Fox, the content is derived from a number of resources on the Web, such as web pages, images and other resources identified by Uniform Resource Locators (URLs) that identify their locations on the Internet (“**a communications network**”).

- (a) “**allowing the selection of at least one resource from a plurality of accessible resources via said communications network, wherein each of said resources is associated with a corresponding identifier displayed on a first section of said user’s display;**”
(Claim 63[a])

256. This limitation is disclosed by the combination of Benedict and Fox. With respect to Benedict, it discloses allowing the selection of at least one resource from a plurality of resources. More specifically, Benedict discloses a “Slide Sorter” feature that displays icons, one icon for each slide in an existing presentation:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2



(Benedict, p.44, Fig. 2.18; *see also id.*, p.43 (“Figure 2.18 shows the Slide Sorter displaying five slides. The Slide Sorter represents these slides as scaled down images.”).) Benedict confirms that each of the slides in a presentation can include one or more “**accessible resources**,” such as graphic files and sound files. (Benedict, p.456 (“In Harvard Graphics, each slide in a presentation can have an associated sound (.wav voice or .mid Musical Instrument Digital Interface MIDI file). When the slide is displayed in a ScreenShow, Harvard Graphics plays the sound file for the viewers to listen to.”), p.377 (“You can import bit maps and images in a variety of graphics formats into a slide in the Slide Editor. A *bit-mapped file* is a picture stored in a file, much the same way a photograph is stored on film. An

image file is set of instructions that tell the program how to draw a picture.”) (italics in original).)

257. Each resource is also “**associated with a corresponding identifier displayed on a first section of said user’s display,**” which in Benedict corresponds to an icon corresponding to the slide that includes the resource. Each icon provides an “**identifier**” because it visually identifies the slide for the user, and as explained below, can be specifically selected through the user interface. Each icon is also “**displayed on a first section of said user’s display,**” because it appears in a particular portion of the user’s screen. In Figure 2.18 of Benedict above, each icon is displayed as a rectangle in a particular part of the screen for displaying slides for the currently loaded presentation (BENSIG1.PRS).

258. Benedict also discloses “**allowing the selection of at least one resource from a plurality of accessible resources,**” because Benedict allows the user to select a slide by clicking on its corresponding icon image (“**identifier**”) in the Slide Sorter. (Benedict, p.44 (“When you select a slide, a highlight box appears around the slide, indicating that it is active.”), p.248 (“You can select one slide or a group of slides for editing in the Slide Sorter. To select a single slide, click the image of the slide. When you select a slide, Harvard Graphics highlights the image with a box. The slide surrounded by a box is the active slide.”).) The user can thus select

a slide that includes, for example, at least one graphic or sound file (“**at least one resource from a plurality of accessible resources**”). Because the slide contains the one or more resources, as explained, the user selects the one or more resources by selecting the slide through the user interface in Benedict.

259. I note that Benedict does not appear to disclose a system in which the resources are accessible “**via a communications network**,” but this would have been obvious in further view of **Fox**. For the same reasons explained for claim 1, it would have been obvious to combine Fox with Benedict and Shimizu (discussed below for later limitations of 63), predictably resulting in the resources being Web resources, such as web pages, graphics, sounds, etc., identified by URLs and accessible over the Internet. Fox confirms that Web resources identified by URLs would have included the same types of resources that can be incorporated into slides in Benedict, including graphic and sound files. (Fox, p.45 (“The Web combines graphics, video, sound, text, hypertext, and more.”), p.13 (“The World Wide Web gives every article, newsgroup, database, computer, file or other Internet fragment its own URL.”).) As further explained for claim 1, in fact, Fox makes clear that URLs could have been used to identify resources to carry out the same functions as Shimizu and Benedict, *i.e.*, to create sequential presentations. (Fox, p.316.) The motivation to combine was fully explained for claim 1[a], which applies here.

260. I provided a complete detailed description of the motivation to combine Shimizu, Benedict and Fox in my discussion of claim 1[a] above, and those motivations continue to apply here. But additional motivations also exist as applicable to this claim limitation. For example, an ordinarily skilled artisan would have been motivated to apply the Slide Sorter user interface teachings of Benedict to the proposed combination because they offer a convenient way to visualize the sequence of nodes and manipulate their order. Benedict provides an express motivation to combine by explaining that the Slide Sorter “enables you to view many slides simultaneously” (Benedict, p.43), thus allowing the user to more quickly review the nodes. Benedict further explains that with the Slide Sorter, “you can see all the information on each slide... and evaluate the slide with a global view of the information.” (Benedict, p.247.)

261. Conceptually speaking, the Slide Sorter is a computer-based implementation of the longstanding technique, well-known in the art and the general public, of organizing photographic slides by selecting and ordering them for presentation using a photographic 35mm slide projector. (Benedict, p.247 (“The Slide Sorter is analogous to the slide table you use with a slide projector. You place the slides in the projector in the order in which you want the audience to view your slides.”).) An ordinarily skilled artisan would thus have regarded the Slide Sorter

interface in Benedict as an appropriate and user-friendly way of allowing users to organize and order the nodes in the show structure.

262. An ordinarily skilled artisan would also have appreciated that the icon view enhances user experience by providing a “bird’s eye” view of an overall presentation, thus enabling far more efficient editing and manipulation. (Benedict, pp.247-248.) For example, the Slide Sorter interface in Benedict enables the user to easily reorder slides by simply moving a slide to a new position using simple drag-and-drop techniques. (Benedict, pp.44, 251-252.) The proposed combination would thus have provided a clear advantage over a user interface that could only represent one slide on the screen at a given time.

263. As explained for claim 1, an ordinarily skilled artisan would have had a reasonable expectation of success and could have implemented the proposed combination using conventional techniques. Additionally, the ability to represent data items within software (such as nodes under the proposed combination) as user-selectable on-screen objects, such as the Slide Sorter in Benedict, would have required nothing more than utilizing basic user interface programming techniques well known to ordinarily skilled artisans. The user interface techniques recited in claim 63[a] are not complex and could have been implemented using conventional programming techniques.

(b) **“providing an interface for creating a show structure of nodes each node identifying a corresponding resource, by allowing generation of a list of said resources by dragging and dropping said identifiers on a second section of said user’s display;” (Claim 63[b])**

264. Turning first to the **“show structure of nodes,”** as explained for claim 1[a], the proposed combination discloses **“creating a show structure of nodes, each node identifying a corresponding resource,”** because it discloses creating a set of nodes that define their associated content and their links to other nodes. As fully discussed above, the node and link information created in Shimizu identifies a set of nodes, the content resources associated with each node, and the links that specify the order in which the nodes are displayed. It would also have been obvious based on the further combination with Fox, as explained, that the resources could have been web pages, images, sounds, and other Web resources on the Internet identified by URLs.

265. The proposed combination also renders obvious that the show structure would have been created using **“an interface... by allowing generation of a list of said resources by dragging and dropping said identifiers on a second section of said user’s display.”** The interface corresponds to a graphical user interface that allows the user to identify the sequence of nodes that will be presented to the user, each node including one or more resources as mentioned. The show structure

created under the proposed combination, therefore, would have been based on the user selections through this user interface.

266. Turning first to **Benedict**, as explained, the Slide Sorter provides a drag-and-drop feature that allows the user to arrange the slides in a presentation to create the desired sequence:

You move slides in the Slide Sorter by selecting and dragging the slide to a new position. When you select a slide, a highlight box appears behind the slide, indicating that it is active. As you drag the slide, a *slide placeholder*, the image of a hand that points to the new location, appears between slides to indicate the new location of the slide (see fig. 2.19). When you release the mouse button, the slide appears in its new location, and the other slides adjust to make room for the relocated slide.

(Benedict, p.44 (italics in original); *see also id.*, p.43 (“You can change the order of your slides by clicking and dragging a slide to a new location.”); *see also id.*, p.443 (“If you change the order of the slides, Harvard Graphics updates the links automatically.”).)

267. Shimizu discloses similar functionality for specifying the order of nodes and creating new nodes at particular positions in the show structure. With respect to the ability to relocate nodes, Shimizu discloses, in reference to icons **85**

shown in Figure 8, that “a mouse may be operated (in conjunction with a graphical user interface function) to switch the positions of the nodes B and C (i.e., node C placed to the left of node B) in the left justified order. This operation changes accordingly the order of the link information specific to the node A.” (Shimizu, 7:43-48.) Shimizu also discloses that the user can press the “INS” button shown in Figure 8, which allows the user to “select[] the node candidate for insertion from the insertable node candidate menu.” (Shimizu, 9:14-18, Fig. 12.) If the user selects a new NODE-E for insertion, for example, the user can “designate[] the position into which to insert which also allows the user to “designate the position into which to insert the selected node E.” (Shimizu, 9:20-23.) Although Shimizu does not expressly state that drag-and-drop techniques are used to perform these operations, it would have been obvious to implement the node repositioning and new node insertion functionalities of Shimizu via drag-and-drop applied to the displayed node icons, as drag-and-drop techniques were well-known as confirmed by Benedict and further discussed below in the “rationale and motivation to combine” section provided below.

268. The proposed combination therefore discloses and renders obvious creating a show structure of nodes via an interface “**allowing generation of a list of said resources by dragging and dropping said identifiers on a second section of**

said user's display,” as claimed. This is because, under the proposed combination, each “slide” icon in the Slide Sorter, or node icon in Shimizu, would have represented a node in the combined system, each node having one or more corresponding resources. A drag-and-drop feature would have predictably enabled the system under the proposed combination to “**generat[e] a list of said resources by dragging and dropping said identifiers,”** because the sequence used to create the show structure would have been generated in response to drag-and-drop operations on the icons (“**said identifier**”) as described. Each node, as explained, is associated with one or more resources.

269. I note that the claim does not specify any particular format or organization for the claimed “**list.**” A person of ordinary skill in the art would have appreciated that a “**list**” does not require a single data structure stored contiguously in memory. For example, a “linked list” is a well-known data structure in which each node in the list may be stored in different locations and is connected to other nodes via links or pointers. (*See Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.240 (“**linked list** In programming, a list of nodes or elements of a data structure connected by pointers.”).) Each node information structure in Shimizu (which identifies the resource corresponding to the node) is part of a linked “**list**” because it is linked via link information to the node information of other nodes

in the show structure. (Shimizu, 1:61-63, 2:5-9 (“The link information of node A **11**, as illustrated in FIG. 2, has pointers arranged in the list format data constitution connecting the first link information **21** (link #1), second link information **22** (link #2) and third link information **23** (link #3).”).)

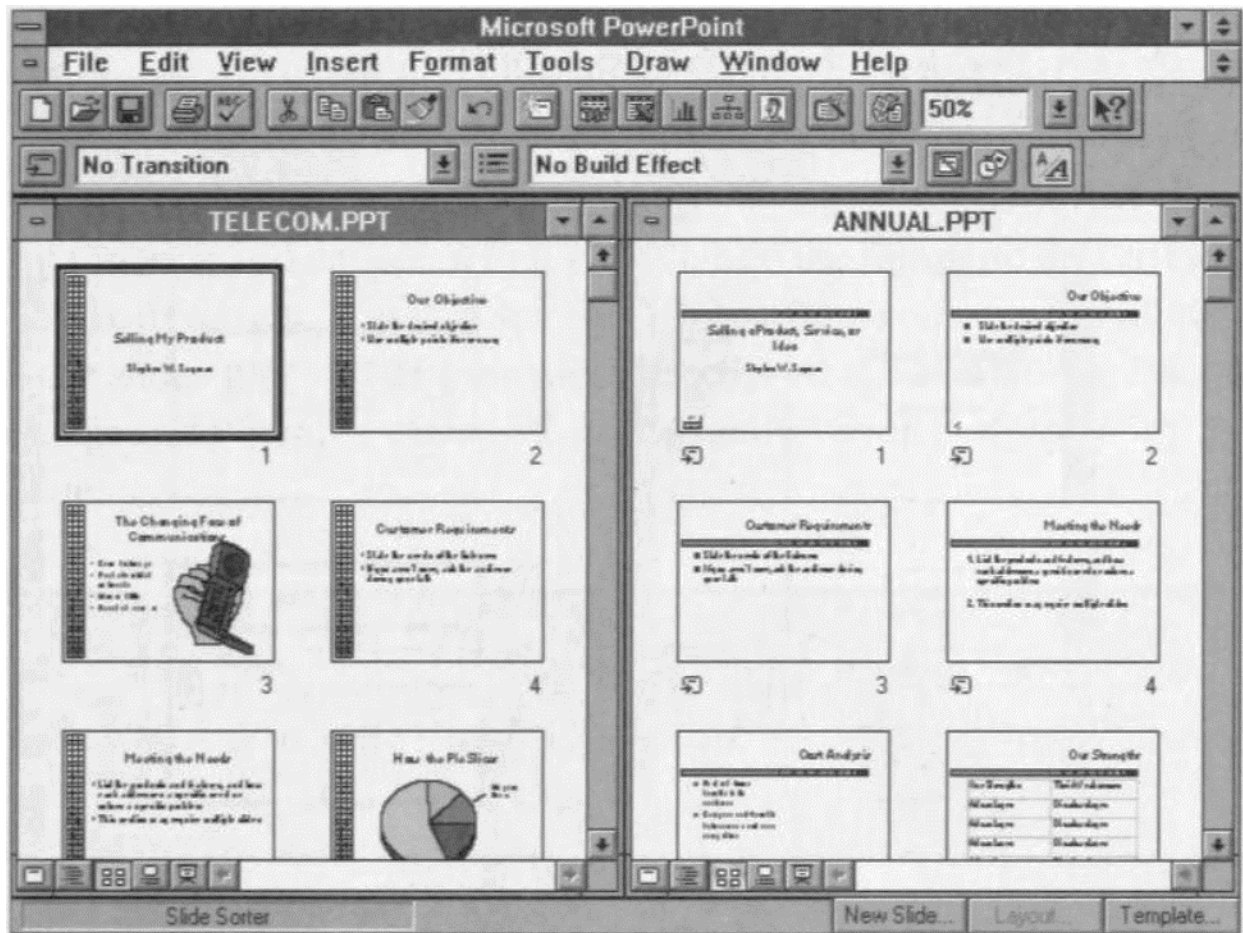
270. I note that it is possible that Patent Owner will argue that Shimizu and Benedict do not render obvious “**dragging and dropping said identifiers on a second section of said user’s display**,” based on the idea that the icons in Shimizu and in the Slide Sorter appear within a single section of the user’s display. I would disagree with this argument because, as explained, the drag-and-drop interface in Benedict allows the user to drag a slide icon from its original location on the screen (“**a first section**”) and move it to another position (“**a second section**”). Similarly for Shimizu, each icon corresponding to a node corresponds to a distinct section of the screen. The claim does not appear to require “**a second section**” that is physically separated or divided from “**a first section**.” I also note that Shimizu discloses a “node candidate menu” for inserting new nodes as discussed (9:11-37), and an ordinarily skilled artisan would have understood that the menu would have understood the menu to represent a different section of the screen than display area **83** that lists the node icons.

271. Nevertheless, even if the claim were interpreted in a narrow fashion that

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

Patent Owner contends is not disclosed by Shimizu or Benedict, such a narrow reading of this limitation would have been obvious based on the further combination with Sagman. Sagman describes a software program known as Microsoft PowerPoint that provided functionality similar, in some respects, to the Harvard Graphics product described above. PowerPoint allowed users to create a presentation including a series of slides. (Sagman, pp.6-7.) Sagman discloses a “Slide Sorter” feature similar to the Harvard Graphics “Slide Sorter,” but with an additional feature – the ability to show two separate side-by-side presentations on the screen such that slides can be moved or copied using drag-and-drop:

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

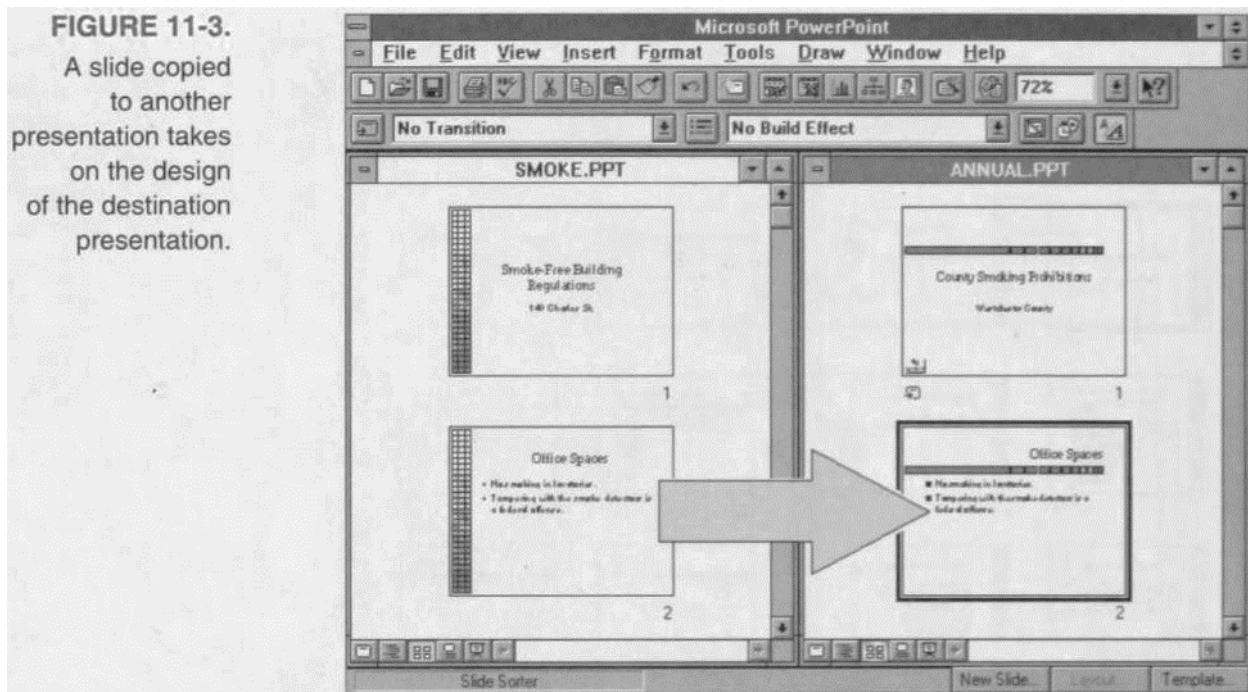


(Sagman, p.311.) The screen display above shows two PowerPoint presentations (TELECOM.PPT and ANNUAL.PPT) displayed side-by-side in two windows, with icons representing the individual slides in each presentation. Sagman explains that, using standard drag-and-drop techniques, the user can drag a slide from one presentation into the other:

After the presentations are arranged side by side, you can move a slide from one presentation to another by clicking on a slide in one presentation and dragging it to the other presentation. The moved slide

takes on the design of the destination presentation's template. To copy a slide rather than move it, hold down the Ctrl key as you drag the slide to the destination presentation. In Figure 11-3 on the next page, the bottom slide has been copied from one presentation to another. Note its appearance in each presentation.

(Sagman, p.313.) Figure 11-3, referenced above, shows the result of copying a slide from one presentation into another:



(Sagman, Fig. 11-3, p.312.) In this example, the slide “Office Spaces” was copied from SMOKE.PPT into the second position of the ANNUAL.PPT presentation. As shown, the slide in the target presentation retains the same content as the original slide but takes on the design of the target presentation.

272. *Rationale and Motivation to Combine (Shimizu, Benedict and Fox with Sagman):* It would have been obvious to combine the system based on Shimizu, Benedict, and Fox, with the above-described disclosures from Sagman. This would have predictably resulted in the Shimizu-Benedict-Fox system in which the “**show structure of nodes**” would have been created using an interface allowing the user to create a show structure by dragging-and-dropping icons (“**identifiers**”) representing nodes from an existing show structure (in “**a first section of said user’s display**”) into the show structure being created (to “**a second section of said user’s display**”). This would thus have predictably resulted in “**creating a show structure of nodes each node identifying a corresponding resource, by allowing generation of a list of said resources by dragging and dropping said identifiers on a second section of said user’s display,**” using these techniques, because the user would have dragged the icons to the desired position in the show structure being created. The drag-and-drop operation “**generates a list of resources**” because it specifies a sequence of nodes, each node identifying one or more resources. Under the proposed combination with Shimizu, as explained previously, each resource would have been identified by URLs and recorded in the node information of its corresponding node as explained. (Shimizu, 5:55-59 (“The entity information pointer slot **63** retains the pointer pointing to the entity information to be displayed

by the node in question. The entity information thus pointed to may be a character string, image information, motion pictures or sounds.”).)

273. Like Benedict discussed above, Sagman is also an analogous reference in the same field of techniques for providing organized or sequenced presentation of content, and would have been reasonably pertinent to a number of problems facing the inventor. For example, Sagman would have been reasonably pertinent to the problem of how to design a user interface to design a show sequence of nodes (each node having corresponding resources), and how to use known drag-and-drop techniques to allow the user to carry out these functionalities using conventional user interface techniques.

274. An ordinarily skilled artisan would have been motivated to adapt this feature from Sagman to the Shimizu-Benedict-Fox combination in order to improve user experience. This is because dragging and dropping nodes (slides) from one presentation into another would have facilitated a more rapid creation of the claimed **“show structure of nodes.”** Sagman also provides an express motivation to combine by explaining that this drag-and-drop technique “lets you transfer an existing slide to a new presentation rather than recreate its content.” (Sagman, p.312.) This would have enabled the user to more rapidly create a new show structure of nodes by moving or copying existing nodes from another show structure

into the show structure being created.

275. Like Sagman, Benedict discloses the ability to move or copy slides from one presentation to another but uses Cut, Copy, and Paste functions through on-screen menus instead of drag-and-drop techniques. (Benedict, pp.252-255.) An ordinarily skilled artisan would have been motivated to improve the proposed combination by using drag-and-drop to perform this function. The ability to drag-and-drop objects was a standard feature of conventional graphical user interfaces and employed in many applications such as the Microsoft Office suite and Harvard Graphics as explained. Sagman further confirms that it was well-known to drag-and-drop objects between different windows on the same screen.

276. Drag-and-drop techniques also had advantages over other known user interface techniques (such as pop-up or pull-down menus). For example, drag-and-drop techniques allowed users to move on-screen objects using gestures that more closely mimic how human beings manipulate objects in the real world, which made drag-and-drop techniques exceedingly easy to learn and use and a particularly desirable choice for user interface design. An ordinarily skilled artisan would have been well aware of these benefits.

277. An ordinarily skilled artisan would have had at least a reasonable expectation of success, and could have implemented the proposed combination using

conventional programming techniques. Techniques for creating and displaying icons on a user interface, and for allowing the user to manipulate and reposition those icons using drag-and-drop techniques, was well-known and a standard fixture of popular operating systems by the mid-1990s, such as Microsoft Windows, IBM OS/2, Apple Macintosh OS, the X Window system available for Unix, among others. These operating systems also offered standard and well-documented APIs for accessing these drag-and-drop features and incorporating them into applications. The disclosures of Benedict and Sagman would have provided more than sufficient details to have enabled an ordinarily skilled artisan to implement the user interface features I am citing for this claim limitation, without undue experimentation.

- (c) **“without requiring user input, automatically accessing each of said resources identified by each of said nodes from said network accessible resources in accordance with said show structure;” (Claim 63[c])**
- (d) **“automatically retrieving content corresponding to each of said accessed resources; and” (Claim 63[d])**
- (e) **“causing the presentation of said retrieved content automatically accordance with said show structure such that during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.” (Claim 63[e])**

278. Claim 63[c]-[e] are substantially similar to claim 1[b]-[d]. As

explained for claim 1 [b], in connection with the display of the show structure under the proposed combination, each of the resources identified by each of the nodes would have been accessed and “**automatically**” and “**without requiring user input.**” Under the proposed combination with Fox, as explained for claim 1, the resources would have included “**network accessible resources,**” *e.g.*, Web resources accessible via the Internet. The step of accessing would also have occurred “**in accordance with said show structure,**” as explained, because the resources would have been accessed based on the sequence specified by the corresponding link information for the nodes. (Shimizu, 8:67-9:4, 5:11-14.)

279. As further explained for claim 1 [c]-1 [d], because these resources (*e.g.* files, web pages, etc.) contain the underlying content, by accessing and retrieving these the resources, “**content corresponding to each of said accessed resources**” is also automatically retrieved. The content is then automatically presented through the user’s computer. (Shimizu, 8:11-16, 8:49-59; Benedict, p.435 (“Self-running presentations automate your ScreenShow by displaying the slides for a specified length of time before the next slide appears—pressing a key is not necessary.”); Fox, p.316) The content is automatically presented “**in accordance with said show structure,**” because as noted it is presented in the sequence specified by the corresponding link information for the nodes. (Shimizu, 8:67-9:4, 5:11-14.)

280. Claim 63[e] concludes by reciting that “**during said presentation of at least some of said content from said accessed resources, at least some of said other network accessible resources are being accessed.**” This limitation is obvious based on Benedict and Wolfe, as explained for claim 1[d].

2. Claim 64

281. Claim 64 is similar in many respects to claim 26 as shown below (overlapping language shown in underlining):

Claim 64	Claim 26
<u>The method in accordance with claim 63, further comprising the step of providing an interactively variable duration information, representing the duration that corresponding content to said resource is presented.</u>	<u>The method in accordance with claim 25, wherein said step of providing an interactively variable duration information</u> further comprises providing a duration information for each of said accessed resources, <u>representing the duration that corresponding content to said resource is presented.</u>

282. As shown, claim 64 is substantially similar to claim 26, and in fact is arguably broader in that it does not require “**a duration information for each of said accessed resources.**” In any case, the limitations of claim 64 are disclosed and rendered obvious for the reasons provided for claim 26. I also separately explained in my discussion of 1[d] why the claimed step of providing interactively variable duration information would have been obvious.

3. Claim 65

283. Claim 65 depends from claim 63, and further recites “[t]he method in accordance with claim 63, wherein the step of causing the presentation of said retrieved content automatically in accordance with said show structure includes the presentation of content from at least two nodes presented simultaneously.”

284. As explained in the discussion of claims 41-42 above, Shimizu discloses the ability to display the show structure though icons 85 in Figure 8 at the same time as node content in display area 81.

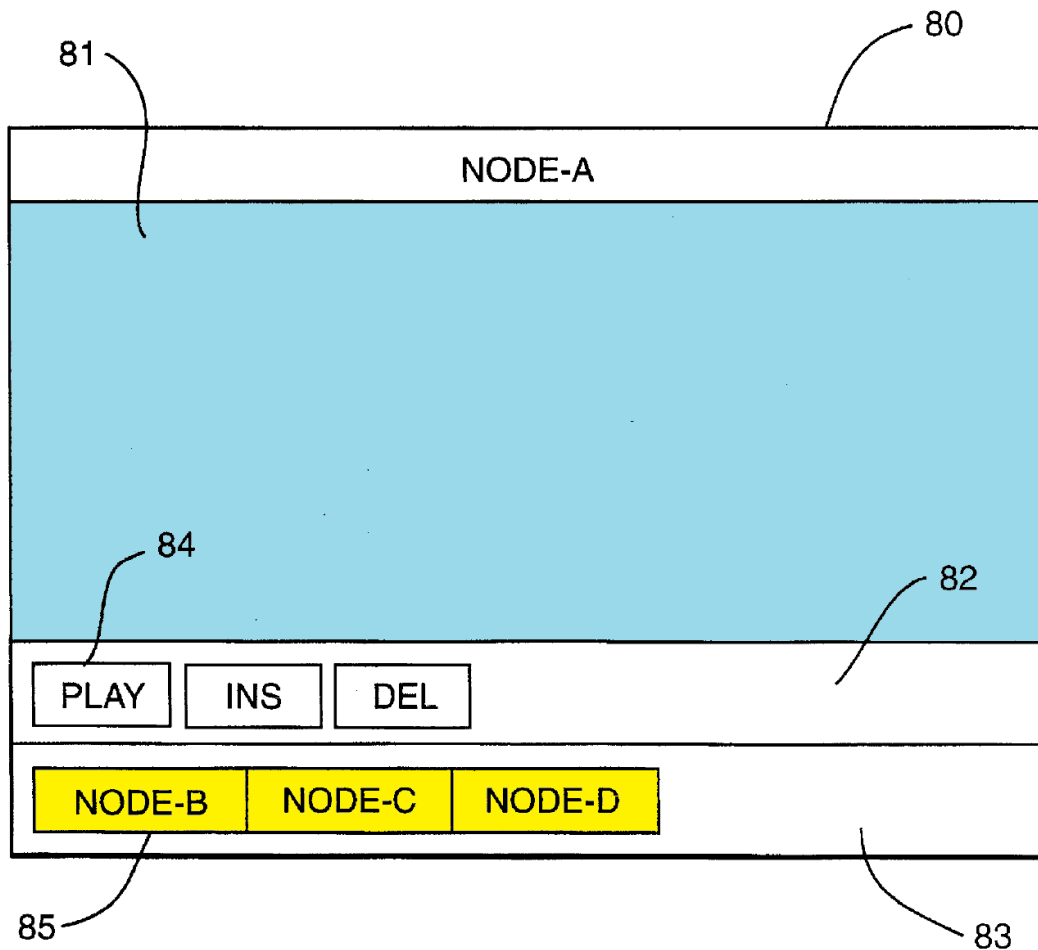


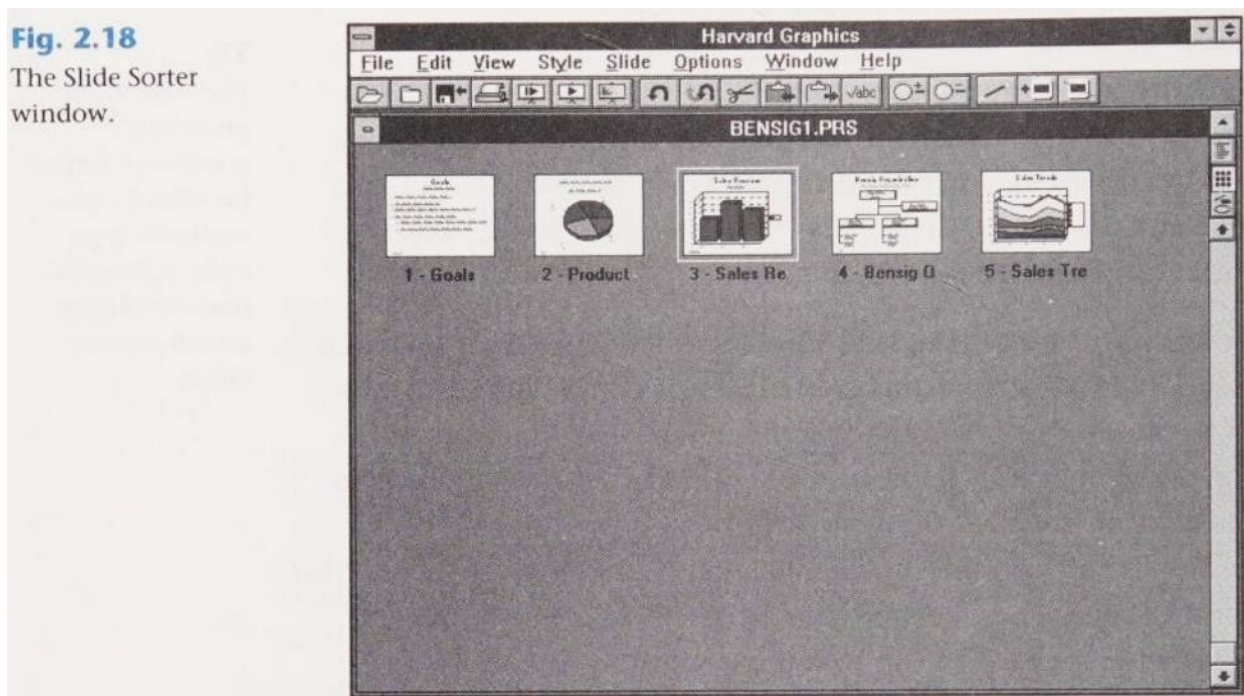
FIG. 8

(Shimizu, Fig. 8 (highlighting added).) As explained for claims 10 and 41-42, Figure 8 also depicts display area **81** for presenting the content of NODE-A, the currently displayed node. (Shimizu, 7:16-18 (“The window **80** is made up of node content display area **81**....”), 7:20-24, 7:62-65, 7:52-59 (“...the content of the acquired information is reproduced and displayed in the node content display area **81** of the window **80**.”).) Figure 8 displays icons **85** for the other nodes and the order in which

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

they are linked to NODE-A.

285. As also explained for claim 63[a], like Shimizu, Benedict displays icons but also shows them as scaled-down versions of the contents of their corresponding slides:



(Benedict, p.44, Fig. 2.18; *see also id.*, p.43 (“Figure 2.18 shows the Slide Sorter displaying five slides. The Slide Sorter represents these slides as scaled down images.”).) Like Shimizu, Benedict discloses the display icons for each slide in the presentation, numbered based on the order in which they will appear.

286. It would have obvious to combine these user interface elements of Shimizu and Benedict, to adapt the display in Shimizu to show each node icon **85**

alongside its underlying content, such as a smaller version disclosed in Benedict.
(Benedict, p.247 (“Even though most text is too small to read, one advantage of using the Slide Sorter is that you can see all of the information on each slide... and evaluate the slide with a global view of the information.”)). The following figure that I created for illustration shows one way in which such a combination could have been implemented:

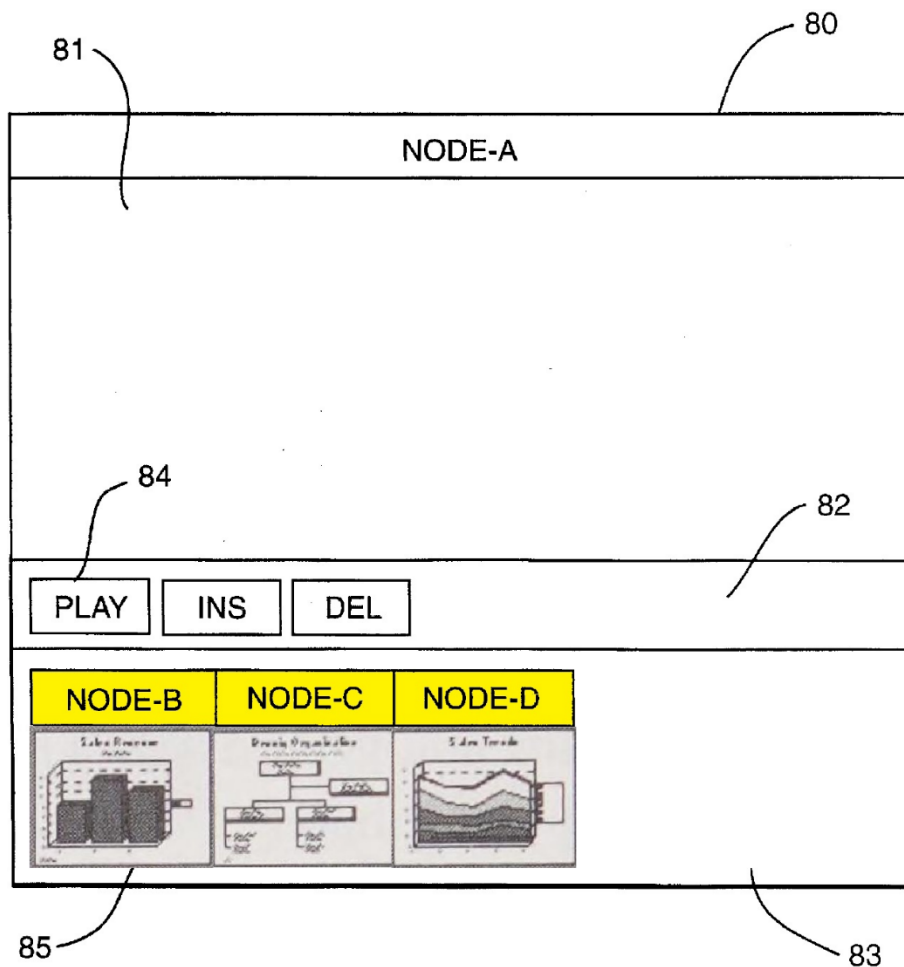


FIG. 8

(Shimizu, Fig. 8 (highlighting added, with scaled down images added).)

287. In the example above, the node icons **85** would have been adapted to also show smaller versions of their content, consistent with the disclosures of Benedict. This would have predictably resulted in the automated presentation of information of claim 63 including **“the presentation of content from at least two nodes presented simultaneously.”** This is because, during presentation of the content of NODE-A as shown in the adapted Figure 8 above, the user can also see content from three other nodes (NODE-B, NODE-C, NODE-D). Although the content of these three additional nodes is smaller, the claim does not require **“presentation of content from at least two nodes”** be full-sized representation.

288. An ordinarily skilled artisan would have found this to be an obvious combination of the disclosures of Shimizu and Benedict, and would have been motivated to combine in this manner to improve user experience. This is because the smaller images in Benedict would have benefitted the system of Shimizu by providing the user with a valuable visual cue and preview regarding content of subsequent nodes in the show structure (such as to better inform the user regarding node navigation, ordering and deletion). (Benedict, pp.247-48.)

4. Claim 66

289. Claim 66 depends from claim 1, and further recites “[t]he method in

accordance with claim 1, wherein said step of accessing further comprises the step of executing an application file residing in the user's computer." This claim adds nothing of patentable significance, as it would have been obvious that the accessing step of claim 63[c] would have involved executing an application file on the user's computer, *e.g.*, an application program that accesses and displays the content associated with each node. Shimizu and Benedict make clear that these functions are performed by an application program on the user's computer. (Shimizu, *e.g.*, 9:63-66 ("This makes it easy to create illustratively an application program allowing the display of node contents to be switched in a desired order and at predetermined intervals on the hypertext apparatus."), 5:40-42 ("FIG. 8 is a view of a typical user interface screen display given by an application program running on the embodiment..."); *see also* Benedict, p.483 ("To install Harvard Graphics on your hard disk, you must use the INSTALL.EXE program, which is on the first disk. This program decompresses the program files to a directory that you specify...").)

290. Under the further combination with Fox, the system would also have included a web browser executing on the user's computer for displaying node content from the Web (*e.g.* web pages, images, etc.). It would have been obvious to an ordinarily skilled artisan that this web browser would have required one or more application files (*e.g.* executable or ".EXE" files) in order to execute the web

browser on the user's computer.

291. It would have been obvious to an ordinarily skilled artisan that an application program such as the ones in Shimizu, Benedict and Fox would have included at least one computer “**file**” residing on the user's computer. (*See Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.164 (“**file** A complete, named collection of information, such as a program, a set of data used by a program, or a user-created document. A file is the basic unit of storage that enables a computer to distinguish one set of information from another.”).)

5. Claim 67

292. Claim 67 depends from claim 66, and further recites “[t]**he method in accordance with claim 66 further comprising the step of generating a data file resulting from said executing step.**” I note that the claim as originally-issued recited “generating a data tile,” which was corrected to “data file” via a Certificate of Correction signed June 9, 2015.

293. It is unclear whether the step of “**generating**” a data file requires that a new file actually be created, as opposed to creating a copy of an existing file. With respect to the latter interpretation, Shimizu discloses generation of a data file resulting from the executing step, *i.e.*, the data file that represents the content associated with the node. Shimizu explains that, when the node contents are being

displayed, the node contents (which are pointed to by the entity information pointer 63) are accessed and obtained. (Shimizu, Fig. 11 (step 107), 8:49-59.) Under the further combination with Fox, the node content would have been identified by a URL and downloaded over the Internet from a remote server in response to a request by application executing on the user's computer. It therefore would have been obvious that the acquisition and retrieval of node content under the proposed combination would have included at least generating a data file, by downloading the content from the Internet as one or more files and then storing the file(s) to the user's computer (either in volatile storage such as RAM or non-volatile storage such as a hard disk). This was also basic and built-in functionality of "cache" and "history" features of conventional web browsers. (Fox, p.82 ("Once you retrieve a Web page, SkipKnot saves it on your hard disk—graphics and all—for as long as you want to keep it."), *see also id.*, p.93 (describing a History List feature showing previously-visited and locally stored web pages).) With respect to the then-popular Netscape web browser, Fox explains that "[w]henver you retrieve a Web page, images are stored in memory. If you return to a page you've already seen, images load much faster due to this *memory cache*." (Fox, p.138 (*italics in original*); *see also id.* p.164 ("Memory cache [for Mosaic]: Speed up page reloading by storing each page that is read.")). It therefore would have been obvious that a data file would have been stored on the

user computer as a result of the executing step, under the proposed combination.

294. In the event “**generating**” is interpreted as requiring creation of a brand new data file, under the further combination with Fox, data files would have been newly-generated as part of the built-in way in which web servers create content in response to requests from web browsers.

295. As explained for claim 28, it was well-known that web servers on the Internet often generated content in response to requests from web browsers, such as a request to access a resource at a particular URL. Fox describes a process using a well-known process known as the Common Gateway Interface (CGI), which allowed web servers to execute scripts that dynamically generated web content on-the-fly in response to a request from the web browser. (Fox, p.391, p.423 (“CGI scripts are used for doing all of the ‘cool’ stuff on the Net. There are... sites that create graphic images on-the-fly.”).) For example, Fox shows an example of a web page that uses a CGI script to generate a web page image representing the current image captured by a camera in a laboratory. (Fox, p.424, Fig. 15-2.) Aside from images, it was well-known (and in fact more common) to use CGI techniques to generate HTML on-the-fly in response to a user request, which is discussed extensively in Fox. (Fox, pp.421-23.) It would have been obvious to generate new content for a wide variety of purposes, such as news, financial, and weather content,

as these are areas where information regularly changes.

296. Under the proposed combination with Fox, therefore, it would have been obvious that the step of “**generating a data file resulting from said executing step**” would have included a web server generating a new data file (*e.g.* an HTML web page file, an image file, etc.) in response to the request from the web browser. For example, if the entity information pointer slot **63** for a particular node of Shimizu under the proposed combination included a URL to a resource on the Web (such as an HTML web page or image), when that node was to be displayed, the server identified by the URL would respond to the request by generating the data file in response to that request. More concretely, using the on-the-fly image example from Fox discussed above, as a result of the request from the web browser in response to a URL, the web server would have dynamically generated a data file (*e.g.*, image file) and transmitted it to the web browser for inclusion on the page.

297. Fox itself provides multiple express motivations to employ this technique, as it explains that “CGI scripts are used for doing all of the ‘cool’ stuff on the Net,” such as creating graphic images on-the-fly, accessing live video feeds, accessing large databases, among other uses. (Fox, p.423.) The use of CGI to generate content on-the-fly for users was well-known, and an ordinarily skilled artisan would have been motivated to use this technique to provide customized and

tailored content to the requesting user – a clear advantage when compared to simply sending the same (static) content to every requesting user. For example, a CGI script generating an image for a requesting user could dynamically generate different images based on the time-of-day, the location and identity of the user, or based on the current input of a device such as a camera, as discussed.

298. It would also have been obvious that these resources would have taken the form of “**a data file**,” as claimed, because files are fundamental units of data storage in any computer system, and are used by web servers and browsers to store resources for the Web. (Fox, pp.46-47 (identifying standard “file formats” for images, text, movies, etc.), p.118 (describing how to view an HTML web page in a web browser and “save the entire HTML file to your hard disk”); *see also Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, p.164 (“**file** A complete, named collection of information, such as a program, a set of data used by a program, or a user-created document. A file is the basic unit of storage that enables a computer to distinguish one set of information from another.”).)

F. Ground 5: Claims 11-16 Are Obvious Over Ground 1 Prior Art With Glaser

1. Claims 11-12

299. Dependent claim 11 depends from claim 1 and further recites “**wherein**

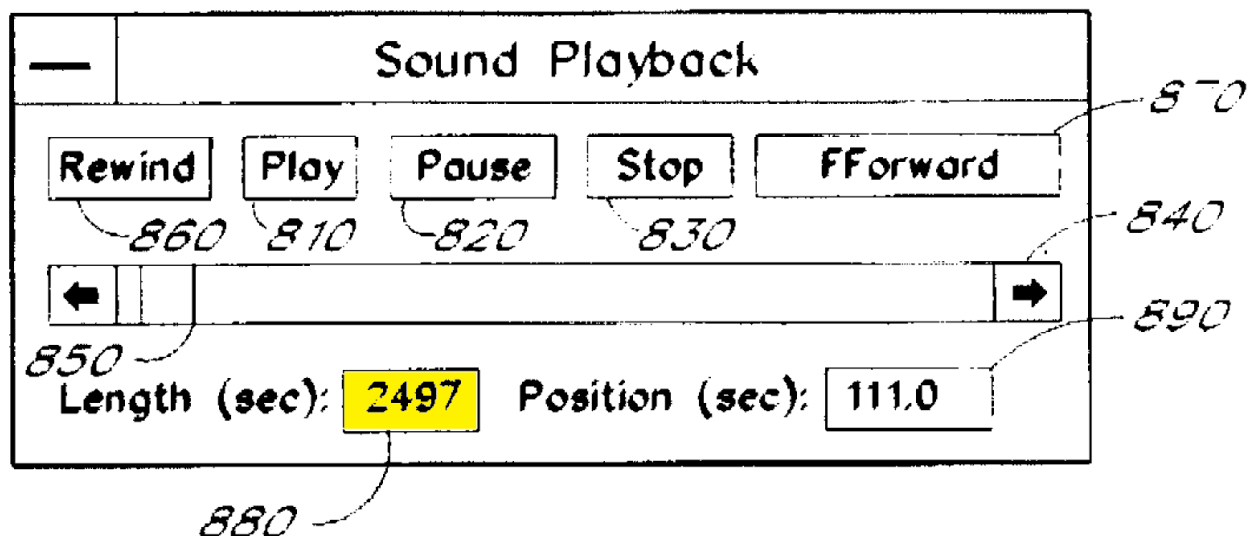
said interactively variable duration information is displayed to the user.” Claim 12 depends from claim 11 and further recites “**wherein said display of said interactively variable duration information to the user occurs during presentation of said content.**” These claims are obvious based on the prior art cited for **Ground 1**, in further view of Glaser.

300. As explained for **Ground 1** above, Benedict discloses an Edit ScreenShow Effects user interface that displays a text box allowing the user to set or change the display time for one or more slides. (Benedict, p.439, Fig. 17.1; *see also id.*, p.440 (“You can set a display time... by typing a time in the Slide display time: text box.”).) Benedict therefore discloses that “**interactively variable duration information is displayed to the user**” because the display time would have been presented to the user, for example, in a text box when the user enters or edits the display time.

301. But Benedict does not disclose that the interactively variable duration information is displayed to the user “**during presentation of said content**” as recited in claim 12. I have therefore cited Glaser in combination with the prior art for **Ground 1** for both claims 11 and 12.

302. Glaser discloses an audio-on-demand system for transmitting audio tracks to a user computer for playback. (Glaser, Fig. 1, 1:54-57, 4:63-5:12.) My

Declaration cites Glaser for a narrow purpose of disclosure of user interface features relating to the playback of content shown in Figure 8A:



(Glaser, Fig. 8A (partial figure; highlighting added).) Figure 8A shows a user interface with control buttons for familiar playback functions including Play 810, rewind 860, Pause 820, Fast Forward 870, and so forth. Slider box 850 allows the user to skip to a particular location within the audio clip and then play the clip from that location. (Glaser, 10:22-30.) Field 890 indicates the amount of time that has elapsed for the audio clip currently being played. (Glaser, 18:19-30.) For purposes of claim 12, the most relevant feature in Figure 8A is length field 880, which displays the length of the audio clip. (Glaser, 19:19-26.)

303. *Rationale and Motivation to Combine (Ground 1 Prior Art with Glaser)*: It would have been obvious to combine the prior art for **Ground 1** with

Glaser. The combination would have predictably resulted in the system based on the proposed combination for **Ground 1**, with the further ability to display the “**interactively variable duration information**” (as described for claim 1[d]) to the user, “**during presentation of said content**,” for example using length field **880** as described in Glaser. In operation, therefore, the system would have displayed to the user how long the content corresponding to the resource will be presented before the system automatically advances to the next node. Although not required by the claim, it would also have been obvious to display further information including field **890** from Glaser indicating how long the current content has been presented.

304. Glaser is an analogous reference in the field of presenting content to a user over a computer network, and would have been reasonably pertinent to problems facing the inventor. For example, Glaser would have been reasonably pertinent to user interface features that allow the user to control playback and determine the length of currently-presented content.

305. An ordinarily skilled artisan would have been motivated to provide this feature in order to provide the user with information as to how long the current content would have been presented before automatic advancement to the next node. This would have provided useful information in allowing the user to maintain his attention. For example, in the case of visually-presented content, suppose the user

reviewed the content quickly and is awaiting automatic advancement to the next node. Without the length field **880** as described in Glaser, the user may not know how long he will have to wait and thus might look away or shift his attention elsewhere. This in turn may have resulted in the user not paying attention when advancement to the next node occurred, causing the user to miss content. Glaser provides a clear benefit by displaying the duration the content will be presented (and optionally how much of that duration has elapsed), allowing the user to efficiently use his time by knowing when the next slide will appear and thus when he needs to refocus attention.

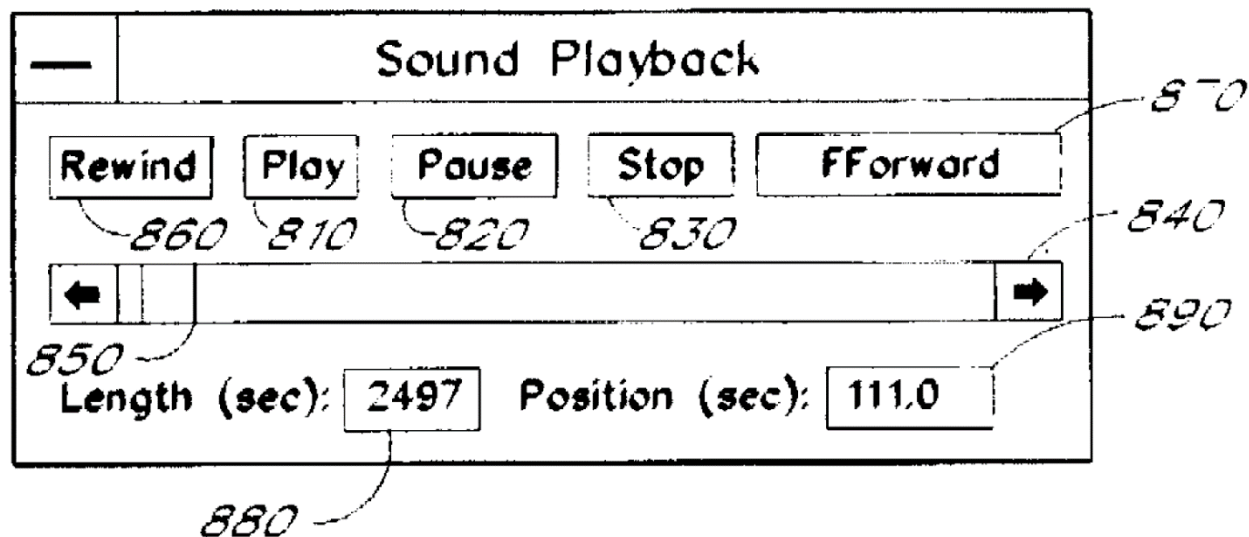
306. An ordinarily skilled artisan would have had at least a reasonable expectation of success. Adapting the proposed combination to display the interactively variable duration information would have been trivially simple to implement. An ordinarily skilled artisan would have found the disclosures of Glaser sufficiently detailed to implement the features discussed above without undue experimentation.

2. Claims 13-16

307. Claim 13 depends from claim 1 and further recites “**providing a play bar to enable the user to vary said duration that said content is being presented.**” Claim 14 depends from claim 1 and further recites “**the step of**

providing control buttons to enable the user to vary said duration that said content is being presented.” Claim 15 depends from claim 14 and further recites “wherein said control buttons provide for pause, fast forward and rewind functions to be applied to said content being presented.” And claim 16 depends from claim 14 and recites “wherein said control buttons provide for skipping of said content being presented.” These features are also obvious in view of Glaser.

308. Referring again to Figure 8A of Glaser:



(Glaser, Fig. 8A; *see also id.*, 11:24-32 (describing Pause field 820), 10:22-33 (describing slider box 850 and Rewind and FForward buttons 860, 870).)

309. Figure 8 shows “a play bar” in the form of a series of user interface buttons including Play 810. The play bar also contains “control buttons” to “enable[] the user to vary the duration” (claims 13-14) that content is presented,

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

for example, through the Fast Forward **870** control that permits the user to reduce the duration of content presentation. Slider box **850**, as explained, allows the user to skip to a later time point within the audio clip being presented. (Glaser, 10:22-33 (seek operation can be performed by slider box **850** as well as “by clicking and holding the mouse button down while the mouse pointer is over rewind or fast forward fields **860**, **870**”).) Glaser also discloses user controls to “**pause, fast forward and rewind**” (claim 15) the content presentation, respectively, through Pause **820**, Fast Forward **870**, and Rewind **860** controls as shown in Figure 8A. These control buttons also “**provide for skipping of said content being presented**” (claim 16), because as explained, the user can through slider box **850** skip to a later location or the end of an audio clip. I also note that Glaser’s user controls are not limited to controlling the presentation of audio content. Glaser discloses that “metadata,” such as text, still images and video, can be associated with audio content and its presentation synchronized with audio content playback. (Glaser, 3:50-58, 23:3-14, 23:54-60, 24:40-67.) An ordinarily skilled artisan would have understood that by, for example, pausing or fast forwarding audio playback, the associated synchronized metadata presentation would also be paused or fast forwarded. (Glaser, 24:52-67 (“Since the metadata advantageously includes a time stamp, it is routine for the server **240** to identify which metadata corresponds to the audio data

after the location designated by the seek message. In this manner, metadata can be provided without delay so that the metadata occurs substantially simultaneously with corresponding audio data.”.)

310. As explained for claims 11-12, an ordinarily skilled artisan would have been motivated to combine **Ground 1** prior art with Glaser to provide these control buttons in order to permit greater control over the presentation of content. Benedict confirms that each of the slides in a presentation can include sound files. (Benedict, p.456 (“In Harvard Graphics, each slide in a presentation can have an associated sound (.wav voice or .mid Musical Instrument Digital Interface MIDI file). When the slide is displayed in a ScreenShow, Harvard Graphics plays the sound file for the viewers to listen to.”).) Moreover, as Glaser itself makes clear, even outside the context of audio playback, the control buttons in Glaser would have been applicable to any kind of content that is presented over a period of time – including node content presented for “**the duration**” under the proposed combination. (Glaser, 3:50-58, 23:3-14, 23:54-60, 24:40-67.) The combination here would have predictably resulted in the system based on the proposed combination for **Ground 1**, with the further ability to display the play bar and control buttons of claims 13-16 to vary the duration of the content being represented.

311. The features shown in Figure 8A, such as pausing, fast forwarding and

213

rewinding would have been widely familiar and to ordinarily skilled artisans and laypersons, as they were derived controls that existed since at least the 1970s with the ubiquitous tape player device. An ordinarily skilled artisan would have been motivated to provide a play bar with these control buttons for the same reasons tape players have for long includes these controls – to give the user greater control over the timing of the presentation of content through simple controls with which the operator is likely already familiar.

G. Ground 6: Claims 30-31 Are Obvious Over Ground 2 Prior Art with Filepp

H. Ground 7: Claims 52-53 Are Obvious Over Ground 3 Prior Art with Filepp

312. Claim 30 depends from 29 and further recites that “**said dynamic content is an advertising message.**” Claim 31 also depends from 29 and further recites “**the step of providing said dynamic content in accordance with a user’s profile.**” Claims 52-53 are identical to claims 30-31, respectively, except that they depend from claims 51-52. As explained, Fox discloses the ability to provide dynamic content as described for claim 29 (**Ground 2**) and claim 51 (**Ground 3**), but does not disclose the additional limitations of claims 30-31 and 52-53.

313. Filepp discloses an “interactive computer system that enables a user to display desired information, such as news, financial and cultural information, and

perform desired transactional services, such as banking and shopping, through any of a plurality of types of personal computers.” (Filepp, Abstract.) Filepp was originally assigned to Prodigy Services Company, an early provider of interactive on-line services that competed with companies such as America Online (AOL) before widespread adoption of the Web.

314. The interactive system in Filepp displays “pages” to the user, with each page divided into a series of partitions as shown in Figure 3a:

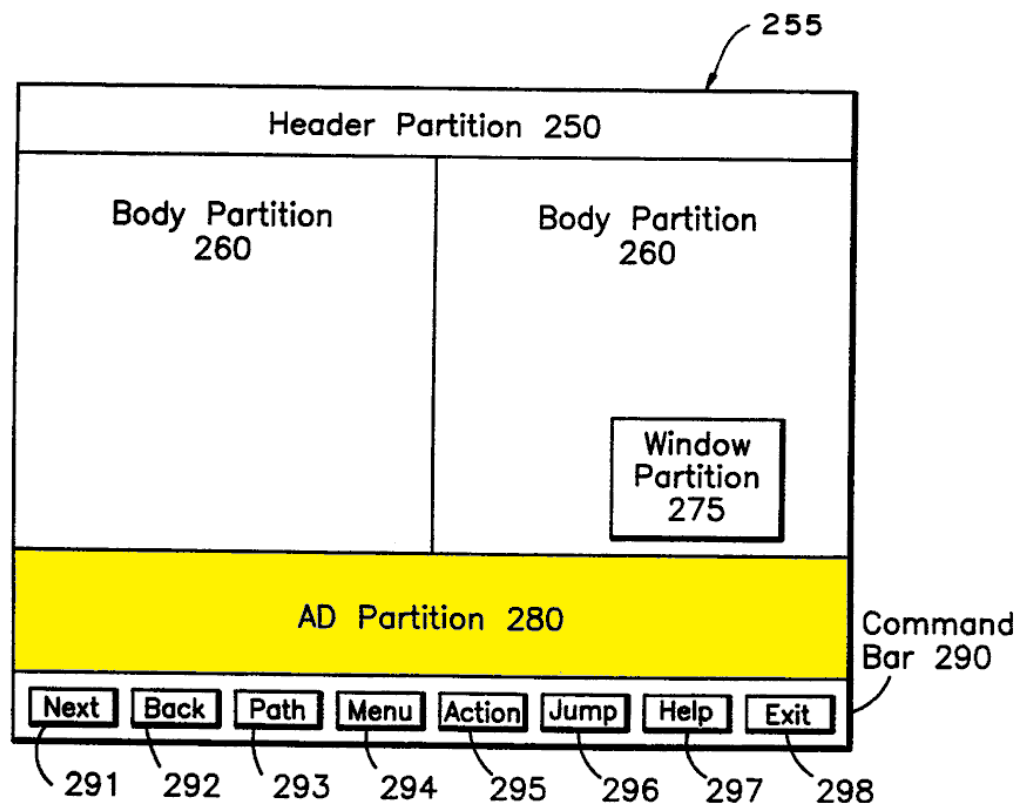


FIG. 3a

(Filepp, Fig. 3a.) “With reference to FIG. 3a, each page 255 is formatted into page

partitions **250**, **260**, **280**, and **290** (not to be confused with applications partitions).” (Filepp, 8:68-9:3.) Header partition **250** “typically conveys information on the page’s topic or sponsor,” and body partitions **260** and window partition **275** “gives the information and transactional content of the page.” (Filepp, 9:10-16.) AD partition **280**, as its name implies, displays advertisements provided over the network for display to the user. (Filepp, 9:27-30, 9:30-34 (“Advertisements **280** may be presented to the user on an individual basis from queues of advertisements that are constructed off-line by business system **130**, and sent to file server **205** where they are accessible to each RS **400**.”), 12:64-66.) Filepp therefore discloses displaying “**an advertising message**,” as recited in claim 30.

315. Filepp also discloses that these advertising messages may be dynamically selected based on information about the user such past activities, demographic information, and other targeting criteria:

Individual queues of advertisements are constructed based upon data collected on the partitioned applications that were accessed by a user, and upon events the user generated in response to applications. The data are collected and reported by RS **400** to a data collection co-application in file server **205** for later transmission to business system **130**. In addition to application access and use characteristics, a variety of other parameters, such as user demographics or postal ZIP code, may be used

as targeting criteria. From such data, queues of advertisements are constructed and targeted to either individual users or to sets of users who fall into certain groups according [sic; to] such parameters.

(Filepp, 9:35-47.) This renders obvious “**providing said dynamic content in accordance with a user’s profile,**” as claimed, because an advertising message (“said dynamic content”) would have been provided in accordance with the user’s interests (based on the previous information they accessed) and/or characteristics (such as demographics, ZIP code, and other criteria). This is consistent with the ’932 patent, which identifies similar information as examples of user profile information. (’932, 8:56-60 (“Such dynamic content is usually based on, among other things, the user’s profile. For example, some of the factors relating to a user profile, which are considered include a particular sports team preference; a demographic; or a pre-set profile as specified in the software.”).)

316. ***Rationale and Motivation to Combine (with Filepp):*** It would have been obvious to combine the prior art for claim 29 (**Ground 2**) and claim 51 (**Ground 3**) with Filepp, in the manner described above. The combination would have predictably resulted in the dynamic content including an “**advertising message**” (claims 30/52) which was provided “**in accordance with a user’s profile**” (claims 31/53).

317. Filepp is an analogous reference in the field of interactive services for displaying content to users, and would have been reasonably pertinent to problems facing the inventor. For example, Filepp would have been pertinent in understanding how to select dynamic content for the user based on its user profile, including advertising content – a feature of the alleged invention.

318. An ordinarily skilled artisan would have been motivated to combine for a simple reason – to make money. It was well-known long before the Internet that companies paid to display advertisements for their products or services across all forms of media (*e.g.* television, radio, newspapers, magazines, billboards, etc.). An ordinarily skilled artisan would have been motivated to adapt the proposed combination to also display advertisement content as disclosed in Filepp. For example, the revenue earned from such advertisements could have been used to defray the costs of developing and deploying the system or for other purposes. Moreover, as explained, advertisements could have been selected based on profile information about the user such as past activities and demographic information, which would have allowed the system to more accurately target products and/or services to the user thus increasing the utility and value of the advertisements.

319. An ordinarily skilled artisan would have had at least a reasonable expectation of success and could have implemented the proposed combination using

conventional programming techniques. It would have been trivial to adapt the proposed combination to include advertisement content in the automatic presentation. The disclosures of the prior art cited above would have provided more than sufficient details to have enabled an ordinarily skilled artisan to implement the proposed combination, without undue experimentation.

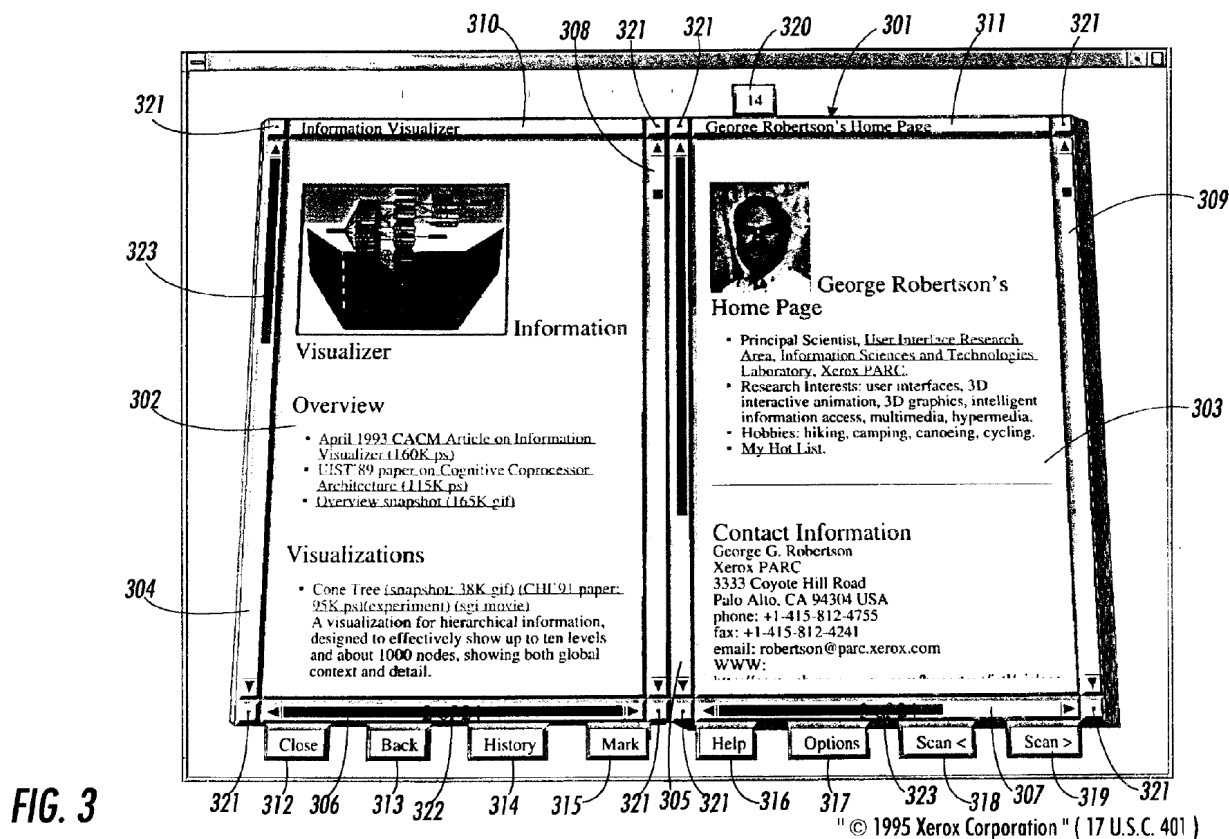
I. Ground 8: Claim 65 Is Obvious Over Ground 4 Prior Art with Robertson

320. In addition to the obviousness ground for claim 65 in **Ground 4**, this claim would have been obvious in further view of Robertson. As noted, claim 65 depends from claim 63 and further recites “**the presentation of content from at least two nodes presented simultaneously.**”

321. Robertson discloses a system for displaying lists of linked documents, such as web pages or other types of linked documents in a hypertext system. (Robertson, 3:24-39.) In one embodiment, Robertson presents a list of web pages by displaying each web page in the list as a “page” of a virtual book. (Robertson, Abstract, 2:14-25.) “The list of pages is displayed in a book metaphor, termed WebBook, on a computer controlled display system... When displayed, two active pages are displayed at one time.” (Robertson, 2:24-28; *see also id.*, 4:14-19 (“The currently preferred embodiment of the present invention presents the list of pages to

a user in a book metaphor hereinafter referred to as a WebBook. The book metaphor is useful since it conveys a relationship between the different pages. Each page in the WebBook represents a page in the list.”.)

322. Figure 3 of Robertson shows an exemplary WebBook view showing two web pages visible in a side-by-side fashion:



(Robertson, Fig. 3.) As explained in Robertson:

FIG. 3 is an illustration of a WebBook as displayed on a display. Referring to FIG. 3, the WebBook 301 is comprised of a first page 302 and a second page 303, each having an identical structure. The first page 302 is an instance of a left page and the second page 303 is an

instance of a right page.

(Robertson, 6:15-19; *see also id.*, 6:28-30 (“The content and layout of the page is determined by the HTML macros defining the page.”).) Robertson explains that, in order to display web pages in a WebBook, each Web page is converted into a page object. (Robertson, 7:65-66.) “Each page is provided in the order in which it is to appear in the WebBook. Accordingly, each page object will represent either a left page or a right page in the WebBook.” (Robertson, 7:66-8:2.) The “Scan >” button 319 in Figure 3 causes the pages in the WebBook to advance forward automatically after a period of time. (Robertson, 6:45-46, 9:5-6.) Another embodiment depicted in Figure 15 shows an enhancement for viewing images representing many or even all of the pages of a WebBook. (Robertson, Fig. 15, 10:15-28.)

323. Robertson thus discloses presenting the content from at least two web pages simultaneously.

324. ***Rationale and Motivation to Combine (Ground 4 Prior Art with Robertson):*** It would have been obvious to combine the references cited for **Ground 4** with Robertson. The combination would have predictably yielded the system based on the proposed combination of **Ground 4** in which “**the step of causing the presentation of said retrieved content automatically in accordance with said show structure includes the presentation of content from at least two nodes**

presented simultaneously.” For example, during the automatic presentation of the retrieved content for the current node, the content from the next sequential node could also have been presented on the display at the same time, as depicted in Figure 3 of Robertson above.

325. Robertson is an analogous reference in the same field of techniques for providing organized or sequenced presentation of content, and would have been reasonably pertinent to a number of problems facing the inventor. For example, Robertson would have been reasonably pertinent to the problem of how to access and display multiple nodes simultaneously, and would have provided useful teachings about how to effectively present linked web pages to the user.

326. An ordinarily skilled artisan would have been motivated to combine in the manner described above in order to improve user experience. As noted previously, Shimizu itself already discloses display of a node and icons **85** for each other linked node in the set of nodes. (Shimizu, Fig. 8, 7:30-35.) An ordinarily skilled artisan would have been motivated to further enhance the proposed combination based on Robertson, as it explains that organizing information using a “book metaphor” was a known way to present information to the user. (Robertson, 1:66-2:3 (“The present invention uses a book metaphor to enable a user to browse through a list of web pages. The use of a book metaphor on computer based systems

is known in the art. Such a book metaphor has been used as an aid for organizing related files and/or applications.”).) An ordinarily skilled artisan would have appreciated that the “book metaphor” of Robertson has a clear advantage of leveraging the user’s familiarity with real-world books to present an intuitive user interface for viewing linked webpages. As an additional advantage and benefit, the techniques in Robertson can display two complete pages at the same time, thus more efficiently utilizing screen space and improving the user’s ability to fully review and visualize the content associated with the nodes.

327. An ordinarily skilled artisan would have had at least a reasonable expectation of success. Robertson itself confirms that web-based techniques, web browsers, and the ability to browse information using a “book metaphor,” were already known to persons of ordinary skill in the art and commercially available. (Robertson, 1:10-2:11.) The techniques in Robertson that I cite for claim 65 could have been implemented using conventional programming techniques, and the disclosures of Robertson would have provided more than sufficient detail to have enabled an ordinarily skilled artisan to implement the cited teachings applied to the proposed combination, without undue experimentation.

VI. NO SECONDARY CONSIDERATIONS OF NONOBVIOUSNESS

328. As explained in **Part III.B**, I understand that “secondary

Declaration of Christopher M. Schmandt in Support of
Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

considerations” may be used as indicia of nonobviousness, provided that any such consideration has a nexus to the invention claimed. I understand from Petitioners’ counsel that Patent Owner in the related district court litigation has not yet identified any evidence related to secondary considerations with a nexus to the claimed invention. Nor am I aware of information, such as commercial success, unexplained results, long felt but unsolved need, industry acclaim, simultaneous invention, copying by others, skepticism by experts in the field, and failure of others, suggesting that the claims addressed in this Declaration are not obvious. To the extent Patent Owner later provides information it claims relates to secondary considerations, I reserve the right to supplement my analysis and opinions to comment on it.

VII. CONCLUSION

329. In signing this Declaration, I recognize that the Declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in this proceeding. If required, I will appear for cross-examination at the appropriate time. I reserve the right to offer opinions relevant to the invalidity of the challenged claims at issue and/or offer testimony in support of this Declaration.

330. I hereby declare that all statements made herein of my own knowledge are true and that all statements are made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001.

Dated: February 21, 2023

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'Chris Schmandt', written over a horizontal line.

Christopher M. Schmandt
Winchester, MA

EXHIBIT A

Christopher Schmandt

4 Longfellow Road
Winchester, MA 01890
+1-617-230-4257

Education

M.I.T., Master of Science, Visual Studies (Computer Graphics), 1980

M.I.T., Bachelor of Science, Computer Science, 1978

Professional Experience -- MIT

Media Laboratory, Principal Research Scientist, 1985-2018 (retired)
Director, Living Mobile Research Group (formerly Speech + Mobility)

Architecture Machine Group, Research Associate, 1980-1984

Architecture Machine Group, Research Assistant, 1979-1980

Architecture Machine Group, Graphics Programmer, 1977-1979

Departmental Undergraduate Research Opportunities Program Coordinator, 1984-2018

Laboratory Intellectual Property Committee 2001-2016, chair 2002-2009

Departmental Committee on Graduate Studies, 1996-2001, 2007-2018

Sponsored Research Activities

Alerting and Mobile Messaging, Digital Life Consortium, MIT Media Lab, 1997-2018

Acoustical Cues to Discourse Structure, National Science Foundation, Principal Investigator, 1995-1998

Parsing Radio, News in the Future Consortium, MIT Media Lab, 1993-1999

Desktop Audio, SUN Microsystems, Inc., Principal Investigator, 1989-1996

Voice Interaction in Hand Held Computers, Apple Computer, Principal Investigator, 1991-1993

Voice Interfaces for Network Services, AT&T, Principal Investigator, 1989-1991

Back Seat Driver, NEC, Principal Investigator, 1988-1991

Acoustic and Visual Cues for Speech Recognition, DARPA, co-Principal Investigator, 1986-1988

Personal Computers and Telephony, NTT Public Corporation, Principal Investigator, 1984-1989

Home Telecomputing, Atari, Inc., Principal Investigator, 1983

Testifying legal engagements within the last five years, representing party in italics

- *Microsoft v. SpeakWare*, 2019
- *Bumble Trading LLC v. Match Group LLC*, 2019
- *Tile Inc. v. Cellwitch Inc.*, 2019
- *Blackberry Ltd v. Facebook, Inc* 2020
- *EROAD Ltd v. PerDiem Co.*, 2020
- *Snap Inc. v. SRK Technology LLC*, 2020
- *Shopify Inc Ltd v. Express Mobile Inc.*, 2020
- *Express Mobile Inc. v Wix Ltd* 2020
- *Motorola Mobility LLC v. Ironworks patents, LLC.* 2021
- *Express Mobile Inc. v GoDaddy Inc* 2021
- *Bumble Trading LLC v. Kinectus, LLC.* 2021
- *Tile, Inc v. Linquet Technologies, Inc.* 2021
- *Facebook, Inc. v. Palo Alto Research Center, Inc.* 2021
- *Unified Patents, LLC v. Gesture Technology Partners, LLC.* 2021
- *Quantum Metric, Inc v. Content Square Isreal, Ltd.* 2021
- *Meta Platforms Inc. v. AlmondNet, Inc* 2022
- *Apple Inc. v. Zentian Limited* 2022

Professional Experience -- Intellectual Property

Co-inventor on U.S. Patents

- 5,177,685 for "Automobile navigation system using real time spoken driving instructions"
- 6,728,348 for "System for storing voice recognizable identifiers using a limited input device such as a telephone key pad"
- 6,937,986 for "Automatic dynamic speech recognition vocabulary based on external sources of information"
- 7,098,776 for "Methods and apparatus for vibrotactile communication"
- 7,392,280 for "Method for summarization of threads in electronic mail"
- 7,738,637 "Interactive voice message retrieval"
- 7,443,283 for "Methods and apparatus for connecting an intimate group by exchanging awareness cues and text, voice instant messages, and two-way voice communications"
- 7,865,560 for "System for summarization of threads in electronic mail"
- 8,121,653 for "Methods and apparatus for autonomously managing communications using an intelligent intermediary"
- 8,135,128 for "Animatronic creatures that act as intermediaries between human users and a telephone system"

Co-inventor on U.S. Patent Applications

- 20020076009 "International dialing using spoken commands"
- 20030023688 "Voice-based message sorting and retrieval method"
- 20030081738 "Method and apparatus for improving access to numerical information in voice messages"
- 20030144846 "Method and system for modifying the behavior of an application based upon the application's grammar"
- 20030158903 "Calendar bar interface for electronic mail interaction"

Publications

Field Study of a Tactile Sound Awareness Device for Deaf and Hard of Hearing Users ISWC 2020. (with D. Jain, B. Chiu, S. Goodman, L. Findlater, J. Froehlich)

SkinMorph: Texture-Tunable On-Skin Interface Through Thin, Programmable Gel ISWC 2018. (with Cindy Kao, M. Banforth, D. Kim)

Technical Interventions to Detect, Communicate, and Deter Sexual Assault. ISWC 2017. (with Manisha Mohan)

Exploring Interactions and Perceptions of Kinetic Wearables. DIS 2017. (with Cindy Hsin-Liu Kao, D. Ajilo, O. Anilonyte, A. Dementyev, I. Choi and S. Follmer)

Leveraging User-made Predictions to Help Understand Personal Behavior. MobileHCI 2017. (with Miriam Greis, Tilman Dangler and Albrecht Schmidt)

Rovables: On-Body Robots as Mobile Wearables. UIST 2016. (with Cindy Hsin-Liu Kao, A. Dementyev, I. Choi, D. Ajilo, M. Xu, and S. Follmer)

DuoSkin: Rapidly Prototyping On-Skin User Interfaces Using Skin-Friendly Materials. ISWC 2016. (with Cindy Hsin-Liu Kao, Christian Holz, Asta Roseway, and Andres Calvo)

Immersive Terrestrial Scuba Diving Using Virtual Reality (with Dhruv Jain, Misha Sra, Jingru Go, Rodrigo Margues, Raymond Wu and Justin Chiu) Proceedings, UIST 2016

Expanding social mobile games beyond the device screen (with Misha Sra) Journal of Personal and Ubiquitous Computing, 2015

NailO: Fingernails as an input surface (with Cindy Hsin-Liu Kao, Artem Dementyev, Joseph Paradiso) CHI 2015

Mugshots: A mug display for front and back stage social interaction in the workplace (with Cindy Hsin-Liu Kao) TEI (Tangible and Embedded Interfaces) 2015

Mime: compact, low power 3D gesture sensing for interaction with head mounted displays (with Andrea Colaco, Ahmend Kirmani, Hye Soo Yang, Nan-Wei Gong, and Vivek Goyal) Proceedings of UIST 2013.

Spotz: A location-based approach to self-awareness (with Misha Sra) Proceedings of Persuasive 2013.

Setting the stage for interaction: A tablet application to augment group discussion in a seminar class (with Drew Harry and Eric Gordon) Proceedings of CSCW 2012.

Indoor Location Sensing using Geo-Magnetism (with Jaewoo Chung, Matt Donahoe, Ig-Jae Kim, Pedram Razavai and Micaela Wiseman) Proceedings of International Conference on Mobile Systems, Applications, and Services (Mobisys) 2011.

My second bike: a TV-enabled social and interactive riding experience (with Jaewoo Chung, Kuang Xu, Andrea Colaco, and Victor Li) Proceedings of IEEE Communications and Networking Conference, Jan 2010.

Going my way?: User-aware route planner (with Jaewoo Chung) Proceedings CHI 2009.

Globetoddler: Designing for remote interaction between preschoolers and their traveling parents (with Paulina Modlitba) CHI 2008 Extended Abstracts

Are we there yet? - a temporally aware media player (with Matt Adcock and Jaewoo Chung), Australian User Interface Conference (AUIC) 2008

Physical embodiments for mobile communication agents (with Stefan Marti), UIST 2005

Giving the caller the finger: collaborative responsibility for cellphone interruptions (with Stefan Marti)
Extended Abstracts, CHI 2005

Active Messenger: email filtering and delivery in a heterogeneous network (with Stefan Marti) Human-Computer Interaction Journal (HCI) Volume 20 (2005)

WatchMe: communication and awareness between members of a closely-knit group (with Natalia Marmasse)
Proceedings of UbiComp 2004

An audio-based personal memory aid (with S. Vemuri, W. Bender, S. Tellex and B. Lassey) Proceedings of UbiComp 2004

Improving speech playback using time-compression and speech recognition (with Sunil Vemuri, Philip DeCamp, and Walter Bender) Proceedings of CHI 2004

Impromptu: managing networked audio applications for mobile users (with Kwan Lee, Jang Kim, and Mark Ackerman), Proceedings of MobiSys 2004

TalkBack: a conversational answering machine (with Vidya Lakshminpathy and Natalia Marmasse) Proceedings of UIST 2003

"ListenIn" to domestic environments from remote locations (with Gerardo Vallejo) Proceedings of the 2003 International Conference on Auditory Display (ICAD)

Safe & Sound: a wireless leash (with Natalia Marmasse) Extended Abstracts, Proceedings of CHI 2003

Mediated voice communication via mobile IP (with Jang Kim, Kwan Lee, Gerardo Vallejo, and Mark Ackerman), Proceedings of UIST 2002.

The Audio Notebook: Paper and pen interaction with structured speech (with Lisa Stifelman and Barry Arons), Proceedings of CHI 2001

Synthetic News Radio (with Keith Emnett) IBM Systems Journal, Vol. 39 Nos. 3-4, pp. 646-659, 2000.

Everywhere messaging (with Natalia Marmasse, Stefan Marti, Nitin Sawhney, and Sean Wheeler) IBM Systems Journal, Vol. 39 Nos. 3-4, pp. 660-677, 2000.

Location-aware information delivery with comMotion (with Natalia Marmasse), Proceedings of the Second International Symposium on Handheld and Ubiquitous Computing, pp. 157-171, Springer, 2000.

Nomadic Radio: Scalable and contextual notification for wearable audio messaging (with Nitin Sawhney), Proceedings of CHI 1999.

Speaking and listening on the run: Design for wearable audio computing (with Nitin Sawhney), Proceedings of International Symposium on Wearable Computing, 1998.

Audio Hallway: A virtual acoustic environment for browsing, Proceedings of UIST 1998.

Dynamic Soundscape: Mapping time to space for audio browsing (with Minoru Kobayashi), Proceedings of CHI 1997.

CLUES: Dynamic personalized message filtering (with Matt Marx), Proceedings of CSCW 1996.

Using acoustic structure in a hand-held audio playback device (with Deb Roy), IBM Systems Journal, Vol 35, Nos. 3 and 4, 1996.

Mailcall: Message presentation and navigation in a nonvisual environment (with Matt Marx), Proceedings of CHI 1996

AudioStreamer: Exploiting simultaneity for listening (with Atty Mullins), short paper, CHI 1995.

Multimedia momadic services on today's hardware, IEEE Network, September/October 1994.

Putting people first: Specifying proper names in speech interfaces (with Matt Marx), proceedings of UIST 1994.

Chatter: A conversational learning speech interface (with E. Ly) AAAI Workshop on Intelligent Multi-Media Multi-Modal Systems, 1994.

Voice Communication with Computers: Conversational Systems. New York: Van Nostrand Reinhold. 1994.

Capturing, structuring, and representing ubiquitous audio (with D. Hindus and C. Horner), ACM Transactions on Information Systems, Vol. 11, No. 4, October 1993.

Speech Recognition Architectures for Multimedia Environments, (with E. Ly and B. Arons), Proceedings of the 1993 AVIOS Conference, September 1993.

Phoneshell: the Telephone as Computer Terminal, Proceedings of the ACM Multimedia Conference, August 1993.

Voicenotes: A Speech Interface for a Hand-Held Voice Notetaker (with L. Stifelman, B. Arons, and E. Hulteen), Proceedings of INTERCHI'93, April 1993.

From Desktop Audio to Mobile Access: Opportunities for Voice in Computing, book chapter in Advances in Human-Computer Interaction, Vol. 4, H.R. Hartson and D. Hix editors. 1992.

Ubiquitous Audio: Capturing Spontaneous Collaboration (with D. Hindus), Proceedings of CSCW'92, November 1992.

Integrating Audio and Telephony in a Distributed Workstation Environment (with S. Angebrannndt, R. Hyde, D. Luong, and N. Siravara), Proceedings of the Summer 1991 USENIX Conference, June 1991.

Augmenting a Window System with Speech Input (with M. Ackerman and D. Hindus), Computer, IEEE Computer Society, Vol. 23, No. 8, August 1990.

Observations on Using Speech Input for Window Navigation (with D. Hindus, M. Ackerman, and S. Manandhar), Proceedings, Human-Computer Interaction, Interact '90, IFIP, August 1990.

Phonetool: Integrating Telephones and Workstations (with S. Casner), Proceedings, GLOBECOM '89, IEEE Communications Society, November 1989.

Desktop Audio (with B. Arons), UNIX Review, October 1989.

Synthetic Speech for Real Time Direction-Giving (with J. Davis), IEEE Transactions on Consumer Electronics, IEEE, September 1989.

The Back Seat Driver: Real Time Spoken Driving Instructions (with J. Davis), Proceedings, IEEE Vehicle Navigation and Information Systems Conference, IEEE, Toronto, Canada, September 1989.

An Audio and Telephone Server for Multi-media Workstations (with M. McKenna), Proceedings, Second IEEE Conference on Workstations, IEEE, Palo Alto, CA., 1988.

Employing Voice Back Channels to Facilitate Audio Document Retrieval, Proceedings, ACM Conference on Office Information Systems (COIS), Santa Clara, CA, 1988.

Conversational Telecommunications Environments, Proceedings, Second International Conference on Human-Computer Interaction, 1987.

Understanding Speech Without Recognizing Words, Proceedings, American Voice Input/Output Society Conference, AVIOS, 1987.

A Robust Parser and Dialog Generator for a Conversational Office System (with B. Arons and C. Simmons), Proceedings, American Voice Input/Output Society Conference, AVIOS, Palo Alto, CA, 1987.

Integrated Messages and Network Services for a Personal Workstation, IEEE Workshop on Telematics and Message Handling Systems, IEEE, 1986.

Voice Interaction in an Integrated Office and Telecommunications Environment, Proceedings, American Voice Input/Output Society Conference, AVIOS, San Francisco, CA, 1985.

Voice Communication with Computers, book chapter in *Advances in Human-Computer Interaction*, H. R. Hartson ed., 1985.

Speech Synthesis Gives Voiced Access to an Electronic Mail System, *Speech Technology*, Vol. 2, No. 3, Aug/Sept 1984.

A Conversational Telephone Messaging System (with B. Arons), *IEEE Transactions on Consumer Electronics*, IEEE, Vol CE-30, August 1984.

Phone Slave: A Graphical Telecommunications Interface (with B. Arons), Proceedings, Society for Information Display International Symposium, SID, San Francisco, CA, June 1984.

Input/Display Registration in a Stereoscopic Workstation, *Displays*, April 1984.

Remote Access to Voice and Text Messages, Proceedings, American Voice Input/Output Society Conference, AVIOS, Washington D.C., 1984.

Fuzzy Fonts: Analog Models Improve Digital Text Quality, Proceedings, National Computer Graphics Association Conference, Chicago, IL, 1983.

Greyscale Fonts Designed From Video Signal Analysis, Society of Applied Learning Technology, Houston, TX, 1983.

Spatial Input/Display Correspondence in a Stereoscopic Computer Graphic Work Station, Proceedings, ACM/SIGGRAPH, Detroit, MI, 1983.

A Programmable Virtual Vocabulary Speech Processing Peripheral (with W. Bender), Proceedings, American Voice Input/Output Society Conference on Voice Data Entry Systems Applications, AVIOS, 1983.

The Intelligent Voice Interactive Interface (with E. A. Hulteen), Proceedings, Human Factors in Computer Systems, National Bureau of Standards/ACM, Gaithersburg, MD, 1982.

Interactive Three-Dimensional Computer Space, Proceedings, SPIE Conference on Processing and Display of Three-Dimensional Data, SPIE, San Diego, CA, 1982, Vol. 367.

Speech Communications, a Systems' Approach, Proceedings, American Voice Input/Output Society Conference on Entry Systems Applications, 1982.

Voice Interaction: Putting Intelligence into the Interface, Proceedings, IEEE International Conference on Cybernetics and Society, IEEE, Seattle, WA, 1982.

The Intelligent Ear: A Graphical Interface to Digital Audio, Proceedings, IEEE International Conference on Cybernetics and Society, IEEE, Atlanta, GA, 1981.

Soft Typography, Information Processing 1980, IFIPS, S. Lavington ed., North-Holland Publishing Co., 1980.
