

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,
Petitioner

v.

ROBOCAST, INC.,
Patent Owner

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

Title: AUTOMATED CONTENT SCHEDULER AND DISPLAYER

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 8,965,932 B2**

(PETITION 1 OF 2)

Table of Contents

	Page
I. MANDATORY NOTICES UNDER §42.8(A)(1)	1
A. Real Party-In-Interest under §42.8(b)(1).....	1
B. Related Matters under §42.8(b)(2)	1
C. Lead and Back-Up Counsel under §42.8(b)(3)	2
D. Service Information	3
II. FEE PAYMENT	3
III. REQUIREMENTS UNDER §§ 42.104 AND 42.108 AND CONSIDERATIONS UNDER §§ 314(A) AND 325(D).....	3
A. Grounds for Standing	3
B. Identification of Challenge and Statement of Precise Relief Requested	3
C. Considerations Under §§ 314(a) and 325(d).....	4
IV. OVERVIEW OF THE CHALLENGED PATENT.....	6
A. Level of Ordinary Skill in the Art.....	6
B. Specification Overview	7
V. CLAIM CONSTRUCTION	9
VI. THE CHALLENGED CLAIMS ARE UNPATENTABLE.....	10
A. Overview of Grounds.....	10
B. Ground 1: Claims 1-11, 14, 16-21 Are Obvious Over Shimizu in View of Benedict, Fox, and Wolfe	11
1. Claim 1	11
(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;” (1[a]).....	13

Table of Contents

(continued)

Page

(b)	“without requiring user input, automatically accessing a plurality of said network accessible resources identified by a corresponding node; and” (1[b])	23
(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,” (1[c])	26
(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.” (1[d]).....	30
2.	Claim 2: “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.”	33
3.	Claim 3: “The method in accordance with claim 1, wherein one or more of said nodes comprises a URL.”	34
4.	Claim 4: “The method in accordance with claim 1, wherein each of said nodes of said show structure of nodes identifies a network accessible resource located on a remote computer.”	34

Table of Contents
(continued)

Page

5.	Claim 5: “The method in accordance with claim 1, wherein one or more of said nodes identifies a resource accessible via the Internet.”	34
6.	Claim 6: “The method in accordance with claim 1, wherein one or more of said nodes identifies a resource accessible via an Internet website.”	34
7.	Claim 7: “The method in accordance with claim 1, wherein one or more of said nodes identifies video content.”	35
8.	Claim 8: “... audio content.”	35
9.	Claim 9: “... displayable content.”	35
10.	Claim 10: “The method in accordance with claim 1, wherein said show structure is displayed to the user.”	35
11.	Claim 11: “The method in accordance with claim 1, wherein said interactively variable duration information is displayed to the user.”	37
12.	Claim 14: “The 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.”	38
13.	Claim 16: “The method in accordance with claim 14, wherein said control buttons provide for skipping of said content being presented.”	38
14.	Claim 17: “The method in accordance with claim 1, wherein said computer is a mobile device.”	40
15.	Claim 18: “The method in accordance with claim 1, wherein said content is presented through said user’s computer using a web browser.”	41
16.	Claim 19: “The method in accordance with claim 1, further comprising the step of altering the show structure in response to user action.”	41

Table of Contents
(continued)

	Page
17. Claim 20: “The method in accordance with claim 1, wherein said show structure includes a sequential path of nodes.”	42
18. Claim 21: “The method in accordance with claim 1, wherein said show structure includes a concurrent path of nodes.”	42
C. Ground 2: Claims 63-67 Are Obvious Over Shimizu in View of Benedict, Fox, Wolfe, and Sagman	45
1. Independent Claim 63	45
(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;” (63[a])	46
(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;” (63[b])	50
(c) “without requiring user input, automatically accessing...” (63[c])	57
(d) “automatically retrieving content...” (63[d])	57
(e) “causing the presentation of said retrieved content.....” (63[e])	57
2. Claim 64: “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.”	58

Table of Contents
(continued)

	Page
3. Claim 65: “The 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.”	58
4. Claim 66: “The 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.”	62
5. Claim 67: “The method in accordance with claim 66 further comprising the step of generating a data file resulting from said executing step.”	63
D. Ground 3: Claims 11-16 Are Obvious Over Ground 1 Prior Art With Glaser.....	65
1. Claim 11: “The method in accordance with claim 1, wherein said interactively variable duration information is displayed to the user.”	65
2. Claim 12: “The method in accordance with claim 11, wherein said display of said interactively variable duration information to the user occurs during presentation of said content.”	65
3. Claim 13: “The method in accordance with claim 1, further comprising the step of providing a play bar to enable the user to vary said duration that said content is being presented.”	68
4. Claim 14: “The 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.”	68
5. Claim 15: “The method in accordance with claim 14, wherein said control buttons provide for pause, fast forward and rewind functions to be applied to said content being presented.”	68

Table of Contents
(continued)

	Page
6. Claim 16: “The method in accordance with claim 14, wherein said control buttons provide for skipping of said content being presented.”	68
E. Ground 4: Claim 65 is Obvious Over Ground 2 Prior Art With Robertson.....	71
VII. CONCLUSION.....	74

Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

List of Exhibits

Exhibit No.	Description of Document
1001	U.S. Patent No. 8,965,932 B2 to Damon C. Torres (filed Dec. 21, 2005, issued Feb. 24, 2015) (“932” or “932 patent”)
1002	Declaration of Christopher M. Schmandt (“Schmandt”)
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”)
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”)
1011	<i>Reserved</i>
1012	Excerpts from Ernest Ackermann et al., <i>Using Unix</i> (1994) (“Ackermann”)

List of Exhibits

Exhibit No.	Description of Document
1013	Excerpts from Microsoft Press, <i>Microsoft Office for Windows 95 Resource Kit</i> (1995) (“Resource Kit”)
1014-1015	<i>Reserved</i>
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”)
1018	<i>Reserved</i>
1019	U.S. Patent No. 5,793,980 to Robert D. Glaser et al. (filed Nov. 30, 1994, issued Aug. 11, 1998) (“Glaser”)
1020	Excerpts from <i>Microsoft Press Computer Dictionary</i> (2d ed. 1994)
1021	Declaration of James L. Mullins, PhD (“Mullins”)
1022	Declaration of David Fox (“Fox Decl.”)
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
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

Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

List of Exhibits

Exhibit No.	Description of Document
1029	Complaint filed in <i>Robocast, Inc. v. YouTube, LLC and Google LLC</i> , No. 1:22-00304 RGA (D. Del. filed March 7, 2022)
1030	Proof of Service of Complaint (Google LLC), <i>Robocast, Inc. v. YouTube, LLC et al.</i> , Case No. 1-22-cv-00304 (D. Del.)
1031	Proof of Service of Complaint (YouTube, LLC), <i>Robocast, Inc. v. YouTube, LLC et al.</i> , Case No. 1-22-cv-00304 (D. Del.)

I. MANDATORY NOTICES UNDER §42.8(A)(1)

A. Real Party-In-Interest under §42.8(b)(1)

Google LLC and YouTube, LLC (collectively “Petitioner”) are the real parties-in-interest to this IPR petition.

B. Related Matters under §42.8(b)(2)

The ’932 patent is the subject of the following currently-pending litigations: *Robocast, Inc. v. YouTube, LLC et al.*, Case No. 1-22-cv-00304 (D. Del., filed March 7, 2022), which involves Petitioner, and *Robocast, Inc. v. Netflix, Inc.*, Case No. 1-22-cv-00305 (D. Del., filed March 7, 2022).

The ’932 patent was the subject of a previous IPR petition, *Unified Patents, LLC v. Robocast, Inc.*, Case No. IPR2022-01125, in which institution was denied on January 20, 2023. *See* Paper 13, IPR2022-01125. Petitioner had no involvement in the *Unified Patents* petition. Petitioner is concurrently filing a second Petition requesting IPR of another set of claims of the ’932 patent, which does not overlap with the claims challenged herein. (*See Petitioner’s Explanation of Multiple Petitions*, filed herewith.)

Petitioner is also filing IPR petitions with respect to two related patents in the same family: *Google LLC et al. v. Robocast, Inc.*, Case Nos. IPR2023-00590 & IPR2023-00591 (U.S. Patent No. 7,155,451) and *Google LLC et al. v. Robocast, Inc.*, Case No. IPR2023-00594 (U.S. Patent No. 8,606,819 B2).

C. Lead and Back-Up Counsel under §42.8(b)(3)

Petitioner provides the following designation of counsel.

LEAD COUNSEL	BACK-UP COUNSEL
Heidi L. Keefe (Reg. No. 40,673) hkeefe@cooley.com COOLEY LLP ATTN: Patent Group 1299 Pennsylvania Ave. NW, Suite 700 Washington, DC 20004 Tel: (650) 843-5001 Fax: (650) 849-7400	Andrew C. Mace (Reg. No. 63,342) amace@cooley.com COOLEY LLP ATTN: Patent Group 1299 Pennsylvania Ave. NW, Suite 700 Washington D.C. 20004 Tel: (650) 843-5287 Fax: (650) 849-7400
	Mark R. Weinstein (Admission <i>pro hac vice</i> to be requested) mweinstein@cooley.com COOLEY LLP ATTN: Patent Group 1299 Pennsylvania Avenue NW Suite 700 Washington D.C. 20004 Tel: (650) 843-5007 Fax: (650) 849-7400
	Reuben Chen (Admission <i>pro hac vice</i> to be requested) rchen@cooley.com COOLEY LLP ATTN: Patent Group 1299 Pennsylvania Avenue NW Suite 700 Washington D.C. 20004 Tel: (650) 843-5007 Fax: (650) 849-7400

D. Service Information

This Petition is being served by Federal Express to the correspondence address of record for the '932 patent, 135621 - IPSILON USA, 110 W 40th Street, Suite 2001, New York, NY 10018. Petitioner consents to electronic service at the addresses provided above for lead and back-up counsel.

II. FEE PAYMENT

Petitioner requests review of 26 claims, with a \$48,250 payment.

III. REQUIREMENTS UNDER §§ 42.104 AND 42.108 AND CONSIDERATIONS UNDER §§ 314(A) AND 325(D)

A. Grounds for Standing

Petitioner certifies that the '932 patent is available for IPR and that Petitioner is not barred or otherwise estopped.

B. Identification of Challenge and Statement of Precise Relief Requested

Petitioner requests IPR institution of claims 1-21 and 63-67 based on:

Ground	Claims	Basis for Challenge under §103
1	1-11, 14, 16-21	Shimizu, Benedict, Fox, Wolfe
2	63-67	Shimizu, Benedict, Fox, Wolfe, Sagman
3	11-16	Ground 1 Prior Art with Glaser
4	65	Ground 2 Prior Art with Robertson

Submitted with this Petition is the Declaration of Christopher M. Schmandt (EX1002) ("Schmandt"), a qualified expert. (Schmandt, EX1002, ¶¶1-10, Ex. A.)

C. Considerations Under §§ 314(a) and 325(d)

§314(a): Petitioner has not previously filed an IPR petition with respect to the '932 patent. As noted in **Part I.A**, the patent was the subject of an IPR petition filed by Unified Patents (IPR2022-01125), in which institution was denied. The *Unified Patents* petition does not favor discretionary denial because (1) Petitioner had no role in its preparation or filing, (2) it challenged fewer claims than all sixty-seven that Petitioner challenges across the present Petition and concurrently filed second petition, and (3) there is no overlap between the prior art in the *Unified Patents* petition and the prior art relied upon here.

As noted, Petitioner is concurrently filing two separate IPR petitions against the '932 patent to account for the number of claims for which challenge is sought; the present Petition challenges claims 1-21 and 63-67 whereas the other challenges claims 22-62 (IPR2023-00593). The filing of two non-overlapping IPR petitions against the '932 patent is permissible as explained in Petitioner's Explanation of Multiple Petitions, filed herewith.

With respect to the pending litigation identified in **Part I.A**, it is in early stages with no discovery having occurred. No claim construction or other substantive rulings have issued from the district court. Petitioner intends to move to stay the pending litigation in the event of institution.

§325(d): The two-part *Advanced Bionics* framework considers: **(1)** whether the same or substantially the same art was previously presented to the Office and if so **(2)** whether Petitioner has demonstrated that the Examiner erred in a manner material to the patentability of challenged claims. *Advanced Bionics* does not apply to Shimizu (EX1003), Benedict (EX1004), Fox (EX1005), Wolfe (EX1014), Sagman (EX1007), or Glaser (EX1019), as none of these references was presented during prosecution of the '932 patent. Only one of the prior art references in the proposed grounds here was considered – Robertson (EX1009).

Petitioner cites Robertson solely in **Ground 5** as an additional basis for obviousness of dependent claim 65. Robertson was cited on an applicant-submitted IDS and the Examiner indicated that it was considered. (EX1026, 0436.) But Robertson was not substantively discussed or the subject of any Office Action.

With respect to dependent claim 65, the Examiner issued an Office Action rejecting that claim based primarily on U.S. Patent No. 5,740,549 to Reilly, the primary prior art the Examiner relied upon in rejecting the independent claims. (EX1026, 0555, 0565 (application claim 66).) Patent Owner responded to the Office Action by amending the claims such that all independent claims now require “**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.**” Patent Owner stated in its response that the Examiner agreed during an

interview that the prior art did not disclose this feature. (EX1026, 0647.) Patent Owner thus made no mention of the feature of claim 65, nor did the Examiner mention it in the Notice of Allowability. (EX1026, 0658-0660.)

Petitioner respectfully submits that the Examiner materially erred in not locating Benedict (EX1004) and Wolfe (EX1014), cited in all of the grounds here, which confirm that the claim amendment feature discussed above – which appears to have been the limitation on which the Examiner relied in allowing the claims – was well-known. (Schmandt, ¶¶62-65, 137-142.) The Examiner erred in not issuing an Office Action rejecting the claims based on the references cited here, which provide a basis of rejection more compelling than the prior art and combinations considered during prosecution.

Although the Examiner allowed dependent claim 65 based on features in the independent claims unrelated to claim 65, Petitioner respectfully submits that the Examiner also erred in not relying on Robertson to reject claim 65. As explained for **Ground 4** in **Part VI.E** below, Robertson renders this claim obvious.

IV. OVERVIEW OF THE CHALLENGED PATENT

A. Level of Ordinary Skill in the Art

A person of ordinary skill as of September 3, 1996 (the earliest application to

which the '932 patent claims priority)¹ would have possessed a bachelor's in electrical engineering, computer science, or similar field, with at least two years of experience developing and implementing network-based computer systems, such as systems for storing and retrieving information over the Internet or communicating using the Web. Such a person could also have qualified with more formal education and less technical experience, or vice versa. (Schmandt, ¶¶13-17.)

B. Specification Overview

The '932 patent 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.) 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.)

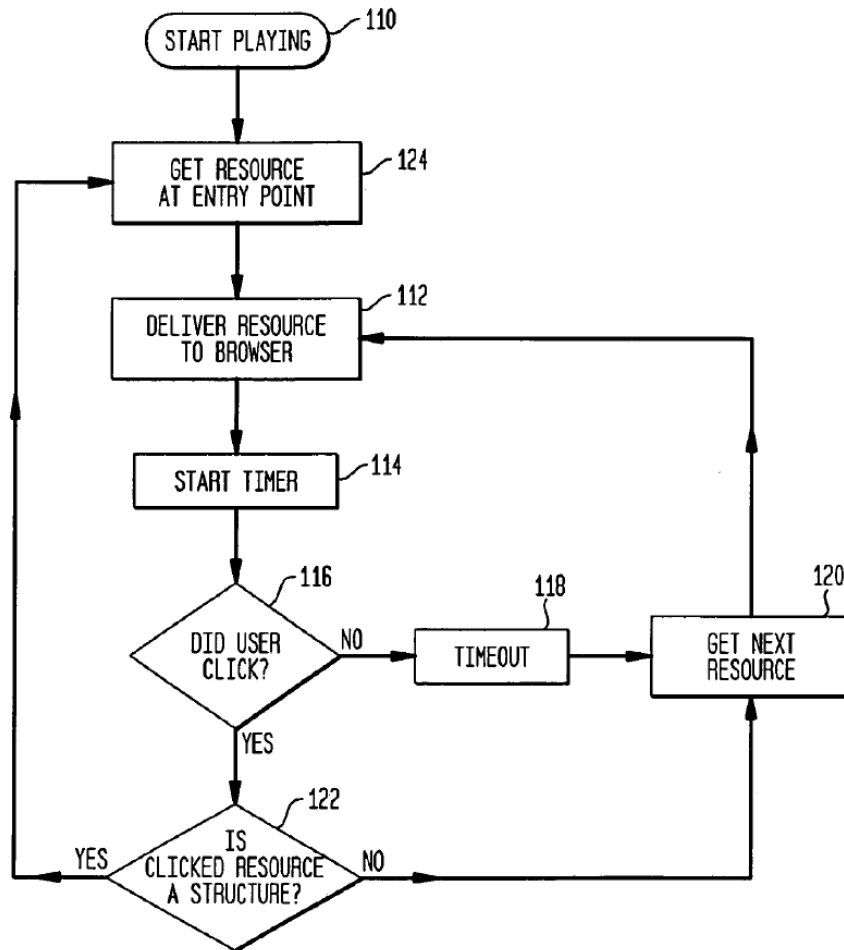
The '932 patent describes methods for a user to be automatically presented with content of a series of resources, such as web pages, using what the patent calls a “show structure.” ('932, 2:59-63.) A “show structure” contains a set of nodes, each node “represent[ing] a resource such as a web page, an executable file, or a data file stored at a predetermined location and accessed via its corresponding address.”

¹ Petitioner does not concede that the challenged claims are entitled to the filing date of the September 3, 1996 provisional application (60/025,360), but that issue is irrelevant as all of the prior art cited here predates that date. Petitioner reserves the right to respond if Patent Owner attempts to argue for an earlier date of invention.

(’932, 2:61-65.) “Each node in the structure 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 is defined by one or more paths that are spanned through these nodes.” (’932, 3:10-11.)

Figure 4 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.)

V. CLAIM CONSTRUCTION

Petitioner does not believe express claim construction is necessary at this time. As discussed in **Part VI** below, a district court in prior litigation not involving Petitioner issued a claim construction order for related U.S. Patent No. 7,155,451

(which shares a near-verbatim written description with the '932 patent) with respect to certain terms and phrases recited in challenged claims. (EX1023.) The Board, in denying IPR institution in *Unified Patents, LLC v. Robocast, Inc.*, Case No. IPR2022-01125, expressly adopted those constructions. (Paper 13, at 13-15.) The prior art cited herein, as demonstrated below, renders the challenged claims obvious under the district court's constructions or under any other reasonable construction.

VI. THE CHALLENGED CLAIMS ARE UNPATENTABLE

A. Overview of Grounds

The challenged claims attempt to lay claim over techniques for presenting a series of content resources in a sequenced and/or organized fashion. But these techniques were well-known long before the patent, as demonstrated by Shimizu (EX1003), Benedict (EX1004), and Fox (EX1005), which disclose techniques for linking a set of hypertext nodes so they can be automatically displayed in a particular order and at specified intervals. Shimizu, Benedict, and Fox provide the core combination on which the grounds here rest. **Ground 1** adds Wolfe (EX1014) for well-known techniques recited in the independent claims for prefetching content during display of other content. **Ground 2** adds Sagman (EX1006), for well-known drag-and-drop user interface techniques recited in claim 63. **Grounds 3-4** address ancillary features in various dependent claims.

Each reference cited in this Petition qualifies as prior art. Shimizu, Wolfe, Glaser, and Robertson qualify as prior art under at least § 102(e) (pre-AIA) because they were filed before the earliest filing date of the '932 patent. The other references qualify under at least § 102(a)/(b) (pre-AIA) because they were published prior to the patent filing date, as evidenced by their availability in public libraries and being otherwise accessible to the public. (EX1021 (Mullins), ¶¶32-55 (Benedict), 56-75 (Fox), 76-99 (Sagman); EX1022 (Fox Decl.), ¶¶3-9.)

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

1. Claim 1

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 unnecessary to determine whether the preamble is limiting because the prior art renders it obvious.

Shimizu discloses a method of “**providing content... on a user’s computer,**” such as text, pictures, and motion pictures:

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; *see also id.*, Fig. 8 (showing user interface for presenting node content), 8:11-25, 8:49-59.)²

The content is also derived from “**a plurality of resources**” *e.g.*, stored information (*e.g.* in data files) containing the character string, image, motion picture, etc., content. (Schmandt, ¶101; Shimizu, 6:39-42, 6:57-59, 8:52-59.) And as explained in more detail below, it would have been obvious that the content would have been derived from resources on the Web (*e.g.*, web pages, images) identified by Uniform Resource Locators (URLs). (Schmandt, ¶¶102, 108-115; Fox, 0049 (“A *URL* (or *Universal Resource Locator*) is the address of a document, file, or other Internet resource.”) (*italics in original*).) It therefore would have been obvious, as explained in 1[a] below, that content would have been provided on a user’s computer “**via a communications network**,” *i.e.*, the Internet. (Schmandt, ¶102.)

Shimizu also discloses displaying content “**in an organized arrangement**,” because the content of each node is presented in an organized (ordered) sequence. (Shimizu, *e.g.*, 9:1-4 (“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.”), 9:60-66, 5:10-14.) As explained below, the organization and sequencing of nodes is determined by link information in Shimizu.

² Except as otherwise noted, all underlining has been added for emphasis.

- (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;” (1[a])

The “**show structure of nodes**” in the proposed combination corresponds to a set of hypertext nodes and their links or connections to other nodes.

Shimizu discloses a system for defining a set of hypertext nodes and their relationships to other nodes. (Shimizu, 6:30-35, Fig. 10; Schmandt, ¶¶44-50.)

Figure 9 shows a “conceptual” example of a set of nodes (7:49-51):

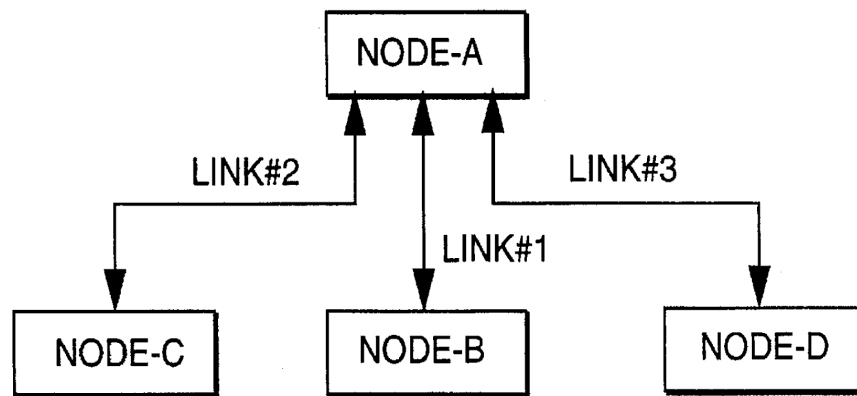


FIG. 9

(Shimizu, Fig. 9.) 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.)

NODE-A is displayed first followed by NODE-C, NODE-B, and NODE-D.

(Shimizu, 7:54-65.) Figure 10 shows a different representation of these nodes:

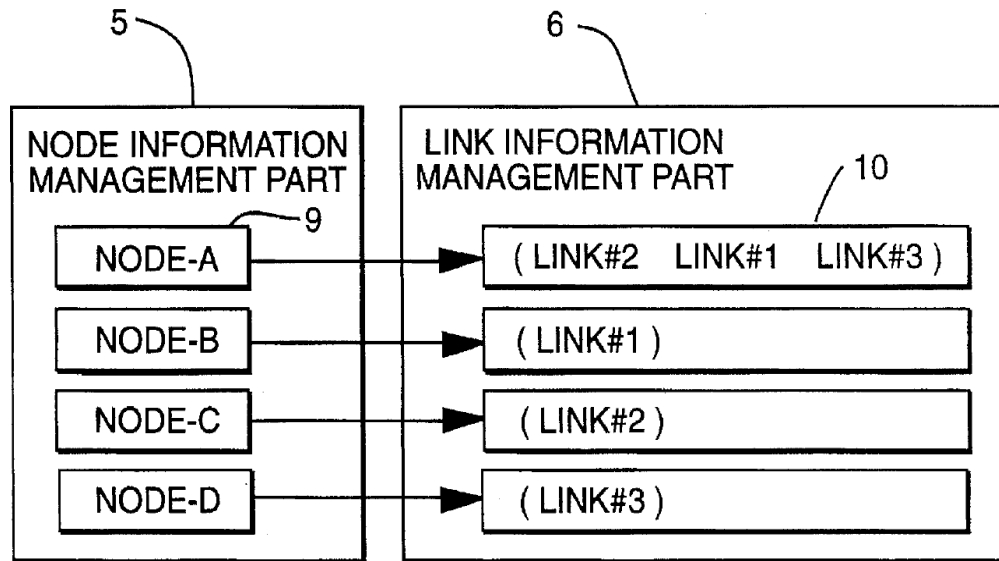


FIG. 10

(Shimizu, Fig. 10, 7:67-8:5.) Node information management part **5**, shown on the left, “manages a plurality of node information structures holding the information about each node.” (Shimizu, 6:31-33.) Link information management part **6**, on the right, “manages a plurality of link information structures defining the relations between the nodes.” (Shimizu, 6:33-35.) In the example above, “the node A is linked to the node C, node B and node D, in that order.” (Shimizu, 8:4-5.)

Shimizu thus discloses creating “**a show structure of nodes**,” which corresponds in Shimizu to an arrangement or set of nodes, which identify their associated content resources and their links to other nodes. With respect to “**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 also would have been obvious that the show structure in Shimizu

was “creat[ed],” as it must have been brought into existence to perform the node display and other functions described in Shimizu. (Schmandt, ¶105.)

Each node in Shimizu also “identif[ies] a resource from a plurality of accessible resources.” As noted, node information is stored in a “node information structure” for each node, as shown in Figure 6:

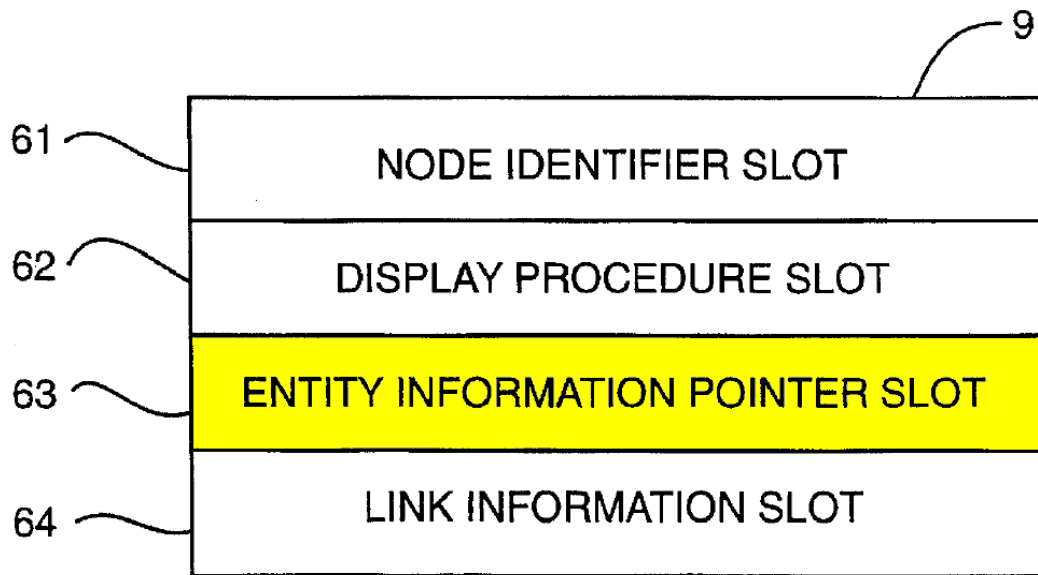


FIG. 6

(Shimizu, Fig. 6 (highlighting added), 6:44-45.) Node identifier slot **61** in Figure 6 “is used to uniquely identify this node information structure within the system.” (Shimizu, 7:1-3.) The key slot here is entity information pointer slot **63** in yellow, which holds a pointer to the content 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 explained for 1[b]-[c] below, Shimizu uses the entity information pointer to identify, retrieve and display content resource associated with the node. (Shimizu, 8:49-59.) Each node in Shimizu thus “**identif[ies] a resource from a plurality of accessible resources,**” because it points to and therefore identifies a particular resource (*e.g.*, entity information such as text, image, motion picture, sound, etc.) from the plurality of resources accessible to the system.

The node information structure shown in Figure 6 also includes link information slot **64**, which stores “the pointer pointing to the link information structure for holding the link information attached to the node in question.” (Shimizu, 6:59-62.) The link information structure for a current node, in turn, points to the node information structure for the next node to which the current node is linked. (Shimizu, 6:63-7:9, Fig. 7.)

It would have also been obvious that “**at least some of which are network accessible resources,**” as recited at the end of 1[a]. Shimizu discloses storing resources in “secondary storage” (*e.g.*, Shimizu, 6:39-43, 2:49-53), which would have been understood as encompassing storage accessible through a network. (Schmandt, ¶107.) Shimizu does not disclose an example in which resources are expressly described as network accessible, but network accessible resources would have been obvious in further view of **Fox**.

Fox explains that the Web is also an example of a hypertext system. (Schmandt, ¶108; Fox, 0044-0046.) Each resource on the Web was identified by a Uniform Resource Locator (URL). (Fox, 0049; *see also id.*, 00332-00333 (describing IMG tag identifying location of image resource).) As was well-known, one of the components of a URL is the domain name of the server on the Internet where the resource is located (*e.g.* “uspto.gov”), so each resource could have been associated with a different website at a unique domain. (Schmandt, ¶110.)

Fox also confirms that it was known to use URLs to identify content resources in the same use case as Shimizu and Benedict, *i.e.*, to display them automatically and sequentially. For example, Fox discloses a technique in which a web page, after a specified amount of time, can cause a web browser to automatically load another web page after 60 seconds, which Fox states could be used to “create an automatic slide show of many Web pages.” (Schmandt, ¶109 (quoting Fox, 00324).) Fox thus confirms that it was known to use URLs to identify resources for a hypertext presentation system such as the Shimizu-Benedict combination in which the content resources associated with a series of nodes are presented in a particular sequence and for a specified duration. (Schmandt, ¶109.)

Rationale and Motivation to Combine (Shimizu with Fox): It would have been obvious to combine Shimizu with Fox, predictably resulting in each node in the claimed show structure being associated with a resource identified by a URL,

such as a web page, image or other Web resource on the Internet. As a result, at least some of the resources “**are network accessible resources**,” *i.e.*, resources accessible over the Internet via a URL. (Schmandt, ¶110.) This would have been a straightforward adaptation; for example, entity information pointer slot **63** of Shimizu could have been readily adapted to hold a URL identifying Web resources on the Internet associated with the node. (Schmandt, ¶110 (citing Fox, 0049 (“The World Wide Web gives every article, newsgroup, database, computer, file or other Internet fragment its own URL.”), 00332-00333 (describing how to use URLs to identify graphic images))).) A URL would have been a natural way to implement entity information pointer slot **63** because a URL provided a standardized way of identifying a file on the Internet. (Schmandt, ¶110.) Shimizu and Fox are analogous references in the same field of organized and sequenced presentation of content and would have been pertinent to problems facing the inventor. (Schmandt, ¶111.)

A skilled artisan would have been motivated to combine Shimizu with Fox to take advantage of the growing popularity of the Web. (Schmandt, ¶¶112-113.) The Background of the '932 patent, in fact, concedes that the Web had already been “popularized in 1993.” ('932, 1:28-31.) Fox similarly describes the enormous commercial potential of the Web and its ability to reach millions of users. (Fox, 00253; *see also id.*, 00224.)

As a related motivation to combine, because nodes under the proposed

combination could have referenced any content on the Internet accessible through a URL, the proposed combination would have benefitted from the ability to draw from a broader universe of available Web content. (Schmandt, ¶114.) A skilled artisan would have had a reasonable expectation of success and could have implemented the combination using conventional techniques. (Schmandt, ¶115.)

Prior District Court Constructions of “Show Structure of Nodes” and Durational Requirement

Before concluding the discussion of 1[a], Petitioner notes that the prior art mapping here is consistent with claim construction rulings in the prior litigation in which the district court construed “**show structure of nodes**” as “*a structure that is arranged for the display of content by specifying one or more paths through a plurality of nodes,*” which must also “*specify the duration of any display.*” (EX1023, 002.) The court further defined “**node**” 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.*” (EX1023, 0016.)

The proposed combination satisfies these constructions. With respect to “**a show structure of nodes**,” the node and link information in Shimizu define a structure that specifies a set of nodes, the content associated with the nodes, and links that specify the path through the nodes. With respect to “**node**,” it would have been obvious that entity information pointer slot **63** stores “*an identifier of a resource that includes an address to the resource,*” because it provides a pointer

identifying the location of the content resource associated with the node. (Schmandt, ¶116.) Under the further combination with Fox, explained above, such resource would have been identified using a URL that provides an address to a Web content resource (*e.g.* web page, image, etc.). (*Id.*)

Petitioner notes that the '932 patent does not require any particular format or organization for the claimed “**show structure**,” but rather, describes it as an arrangement containing a set of nodes. ('932, 2:61-65.) The proposed combination similarly renders obvious an arrangement or set of nodes whose characteristics (including their associated content resources and connections with other nodes) are defined by their node and link information as explained. (Schmandt, ¶117.)

Show Structure Specifying Duration of Display

The prior litigation constructions above further state that a show structure of nodes “*specifies the duration of any display*,” and that a node includes “*the duration for which the resource’s content is to be presented by default*.” These concepts are related to 1[c], separately covered below, reciting “**interactively variable duration information**,” and are obvious under the proposed combination.

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. (Shimizu, 3:9-16, 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.”.) But Shimizu does not appear to disclose whether the predetermined intervals can be changed by a user. But this would have been obvious in further view of **Benedict**.

Benedict describes Harvard Graphics software for creating computer-based slide presentations. (Benedict, 0019, 0041; Schmandt, ¶¶ 52-55, 121.) A user can view a presentation (a “ScreenShow”) that displays slides in a specified order, similar to the sequential display of nodes in Shimizu. (Benedict, 0037.) Benedict discloses an additional feature (known as “HyperShows”) that allowed slides to include hypertext links to jump to other slides. (Benedict, 00123-00134; Schmandt, ¶¶ 121, 174.)

As relevant to the durational requirement, Benedict discloses “ScreenShow Effects that allow the user to customize the display of the presentation.” (Benedict, 00119-00123.) As shown in Figure 17.1 below, one of those features allows the user to specify how long a slide will be displayed before moving to the next slide:

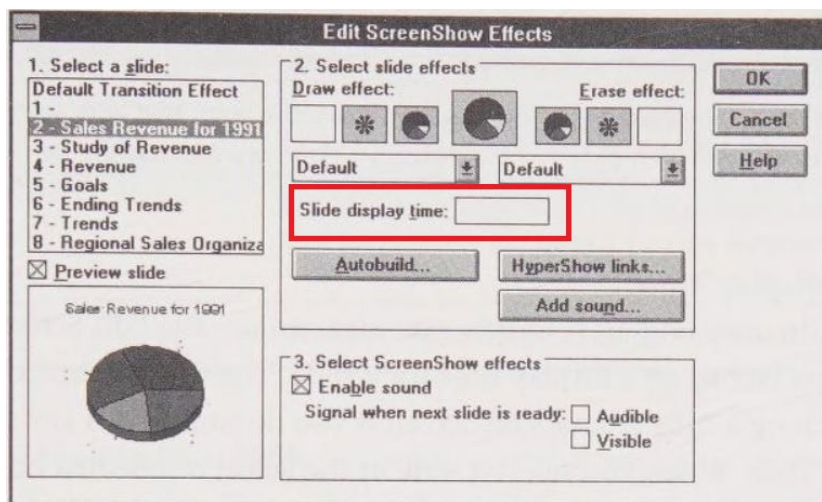


Fig. 17.1
The Edit
ScreenShow
Effects dialog box.

(Benedict, 00120, Fig. 17.1 (red box added).) Figure 17.1 shows editing the ScreenShow Effects for a selected slide (“Sales Revenue for 1991”). The text box shown in red allows the user to specify how long (in seconds) to display the selected slide before automatically advancing to the next slide. (Benedict, 00121 (“The display time determines how long a slide remains on-screen if you do not press a key to move to the next slide....”).) In addition to assigning display time on a slide-by-slide basis (as shown in Fig. 17.1 above), the user can set a display time for all slides. (Benedict, 00122-00123; Schmandt, ¶¶122-123.) Either way, the display time allows the user to create “self-running” presentations that advance to the next slide after a specified period. (Benedict, 00134 (“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.”).)

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 Shimizu-Fox system with the further ability to vary the duration of display of the content resources of each node. (Schmandt, ¶124.) Accordingly, the “**node**” under the proposed combination would have further included information specifying “*the duration for which the resource’s content is to be presented by default.*” (EX1023.) For example, the content resources of a node would have been displayed for a predetermined (default) interval as disclosed in

Shimizu, which the user could have changed in accordance with Benedict. (Schmandt, ¶124.) Benedict is an analogous reference in the same field of organized and sequenced presentation of content and would have been reasonably pertinent to problems facing the inventor. (Schmandt, ¶125.)

As noted, Shimizu itself discloses displaying node content for a “predetermined” time or interval before switching to the next node. (Shimizu, 3:9-16, 5:11-14.) A skilled artisan would have been motivated to enhance Shimizu by allowing the user to further adjust the interval to improve user experience and provide greater customization and flexibility. (Schmandt, ¶¶126-127.) For example, some nodes may have had content that was more complex than other nodes, thus justifying a longer display time to provide the user with more time to view and digest their content. (*Id.*) 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. (Schmandt, ¶128.)

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

This limitation was largely covered in the discussion of 1[a]. Shimizu discloses automatically accessing the content resources identified by the nodes (“**a plurality of... resources identified by a corresponding node**”) for display:

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, 8:13-16.) In order to display the node contents, the entity information pointer slot **63** discussed in the previous limitation is used to identify the location of the content resource, which is then accessed and retrieved. (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.**” Under the proposed further combination with Fox, as noted, the entity information pointer slot **63** would have included a URL associated with the content resources on the Web (*e.g.*, web page, images, etc.) 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 automatically accessed. (Schmandt, ¶129.)

The proposed combination also renders obvious performing the accessing step **automatically** and **without requiring user input**, by proceeding from node-to-node without user intervention. (Shimizu, Fig. 11, (Steps 104-109), 8:67-9:4, 5:11-14.) Shimizu does not disclose (let alone require) any user controls (*e.g.* a “Next” button) to advance to the next node. (Schmandt, ¶130.)

Benedict similarly discloses the “**automatically**” and “**without requiring user input**” limitations because ScreenShows can run automatically. (Benedict, 00134 (“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.”).) And Fox, as noted, also discloses automatically retrieving the next page URL after a specified period of time. (Fox, 00323 (“The server waits and automatically gives the Web page the new information.”).)

Under the proposed Shimizu-Fox-Benedict combination, the user commences automatic display of node content by initially pressing “PLAY” or other control button. (Shimizu, 8:17-19, Fig. 11 (Step **101**).) This initial user step presents no issue because the step of “**accessing a plurality of said network accessible resources identified by a corresponding node,**” is separate from user activation of a PLAY button. (Schmandt, ¶¶133-134.) The district court’s claim construction order, in fact, expressly adopted Patent Owner’s argument that “**without requiring user input**” does not exclude a triggering event by a user causing commencement of the show. (EX1023, 0026-0027; ’932, 2:59-61.)³ The proposed combination thus renders obvious accessing “**automatically**” and “**without requiring user input.**”

³ This is also consistent with Patent Owner’s infringement theory against Petitioner in the pending litigation, which alleges that videos in YouTube playlists proceeded “without requiring input from the computer user beyond (at most) just a single initial click that served to commence a playlist.” (EX1029, 0014, ¶26.)

- (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,” (1[c])**

As just explained for 1[b], the resources associated with each node are automatically accessed and retrieved. Because these resources (*e.g.* files, web pages, etc.) contain the underlying content, by automatically accessing and retrieving these resources, the **“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, 00134 (“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, 00324.) The content is also automatically presented **“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.)

This limitation 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.

This feature recites nothing more than a well-known technique, commonly referred to as “prefetching” or “preloading,” in which a computer loads data before it is actually needed so when the computer later needs that data, it can utilize it immediately. Preloading was a commonly used technique for reducing delays associated with presenting information to a user. (Schmandt, ¶¶62-63, 137-138.)

For example, Benedict discloses a feature in which Harvard Graphics preloaded a subsequent slide during the display of the current slide:

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, 00122 (boldface in original).)

An ordinarily skilled artisan would have understood that the act of accessing information as recited in 1[c] encompasses at least opening a file and reading information from it. (Schmandt, ¶139.) It would have been obvious that, in order to prepare a next slide, the software in Benedict would have accessed its resources by at least reading them into memory. (*Id.*)

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 display of content resources associated with the current node, **“at least some of said other network accessible resources are being accessed,”** *i.e.*, content resources associated with the next node would have been accessed (such as by retrieving them from the Internet via their URLs). An ordinarily skilled artisan would have been motivated to implement this technique to improve the performance of the system and user experience by reducing the amount of delay before display of the next node. (Schmandt, ¶140.)

Benedict does not disclose details about how its next slide preparation technique works. In the event Patent Owner faults Benedict for this lack of detail, therefore, Petitioner has also cited to **Wolfe** which provides a more detailed description of preloading. Wolfe describes a technique in which, while the user views certain information, other information the user 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 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, 2:25-35; *see also id.*, 5:47-59.) Although an embodiment in Wolfe involves an on-line legal database, his techniques are broadly applicable to any type of documents that can be retrieved and presented to the user. (Schmandt, ¶142; Wolfe, 4:20-23.) Wolfe also makes clear that preloading is applicable to networked databases accessible by multiple users simultaneously. (Wolfe, 3:60-4:8.)

Rationale and Motivation to Combine (Shimizu, Fox, and Benedict with Wolfe): It would have been obvious to further combine the Shimizu-Fox-Benedict system with Wolfe, predictably resulting the ability to access “**at least some of said other network accessible resources,**” *i.e.*, content resources on the Web associated with at least 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 display of the current node, the system would have accessed resources (*e.g.*, Web pages, images, etc.) associated with the next node over the Internet. (Schmandt, ¶143.) Wolfe is an analogous reference in the field of content retrieval and would have been pertinent to a number of problems facing the inventor. (Schmandt, ¶144.)

A skilled artisan would have been motivated to combine to reduce or eliminate the delays associated with accessing and displaying the next node. (Schmandt, ¶¶145-146.) Wolfe expressly supports this motivation by explaining that an “undesirably slow” loading process can be reduced by preloading subsequently-

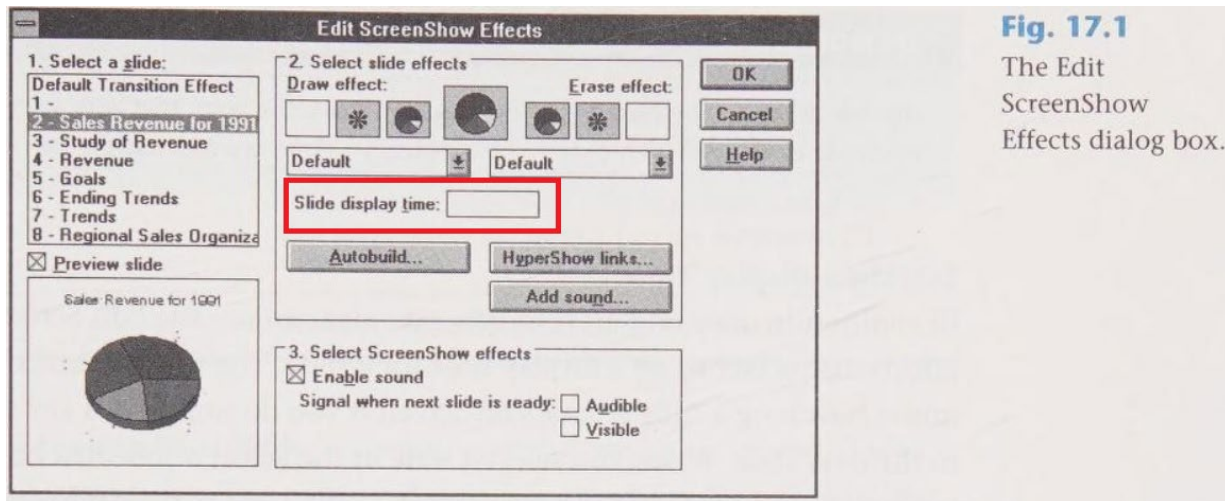
needed information during the display period. (Wolfe, 5:47-59.) As Wolfe explains, “the time that the user spends viewing displayed information is often sufficient to advantageously preload a substantial amount of information.” (Wolfe, 2:31-35.)

Wolfe also explains that preloading is particularly appropriate in a scenario in which the system can predict the information the user will next require. (Wolfe, 2:28-35, 5:52-59.) The proposed combination presents precisely that scenario because nodes are presented (and thus accessed) in the predictable sequence established by the link information. (Schmandt, ¶145; Shimizu, *e.g.*, 9:1-4.) A skilled artisan would have had a reasonable expectation of success and could have implemented this feature using conventional techniques. (Schmandt, ¶148.)

(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.” (1[d])**

It would also have been obvious that the step of creating the show structure of nodes also comprises providing **“an interactively variable duration information.”** Under the proposed combination, each node would have included a display time that could have been interactively varied by the user – under two separate theories. (Schmandt, ¶¶150-154.)

First, as described for 1[a], Benedict discloses that the user can change the display time for one or more nodes using the ScreenShow Effects feature:



(Benedict, 00120, 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, 00121 (*italics in original*).) As noted, this technique can be used to edit the display time for each slide individually, or for all slides. (Schmandt, ¶¶151, 122-123.) The duration information for a node (which specifies how long its content resources will be displayed) can therefore be interactively varied by a user. (Schmandt, ¶151.) The user for purposes of this technique would be the user creating or editing the presentation, who could have been the same user viewing the presentation. (*Id.*)

As a second and independent way of showing “**an interactively variable duration information,**” Patent Owner in the pending litigation appears to be taking the position that this limitation is satisfied by a user’s ability to skip to another node

during display of node content. (EX1029, ¶27.) Petitioner does not concede that this functionality satisfies the claim, but even if it did, a user viewing a node under the proposed combination can simply press a button to immediately advance to the next node. (Benedict, 00121 (“The display time determines how long a slide remains on-screen if you do not press a key to move to the next slide.”), 00124 (“Next slide... Moves to the next slide in the presentation”).) Applying Patent Owner’s own theory, the user in the proposed combination could have shortened the display duration by immediately advancing to the next node before expiration of the automatic display duration. (Schmandt, ¶152.) The rationale and motivation to provide duration information, provided for 1[a], applies here. (Schmandt, ¶¶124-128.)

The district court in the prior litigation 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.*” (EX1023, 0022.) This construction is satisfied by the proposed combination. In the first example above, the parameter corresponds to the display time advancing to the next node, which takes the form of a predetermined (default) value as disclosed in Shimizu, which the viewer can change as disclosed in Benedict. In the second example above, to the extent the claim were construed to encompass skipping functionality, the parameter corresponds to the amount of time the software waits when displaying a current node

before automatically advancing to the next node. (Schmandt, ¶154.)

- 2. Claim 2: “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.”**

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

Claim 2 does not appear to require providing *different* durations for accessed resources, but even if the claim were interpreted in this manner, the user could have specified a display time separately for each node, *e.g.*, on a node-by-node basis as explained. (Schmandt, ¶158; Benedict, 00120, ¶3 (“To see the effects for a particular slide, select the slide from the Select a slide: list box.”); *id.*, 00121 (“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 required by claim 2. (Schmandt, ¶158.) 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. (*Id.*)

3. **Claim 3: “The method in accordance with claim 1, wherein one or more of said nodes comprises a URL.”**
4. **Claim 4: “The method in accordance with claim 1, wherein each of said nodes of said show structure of nodes identifies a network accessible resource located on a remote computer.”**
5. **Claim 5: “The method in accordance with claim 1, wherein one or more of said nodes identifies a resource accessible via the Internet.”**
6. **Claim 6: “The method in accordance with claim 1, wherein one or more of said nodes identifies a resource accessible via an Internet website.”**

These claims are all closely related and can be addressed together. As explained for 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. (Schmandt, ¶160, 110.) 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. (*Id.*; Fox, *e.g.*, 0050 (“The second part of most URLs is the full Internet address of the machine where the file is located.”).)

7. **Claim 7: “The method in accordance with claim 1, wherein one or more of said nodes identifies video content.”**
8. **Claim 8: “... audio content.”**
9. **Claim 9: “... displayable content.”**

As explained for 1[a], each node in Shimizu 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, 0080 (“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 image content, both of which are examples of displayable content. (Schmandt, ¶162.)

10. **Claim 10: “The 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:

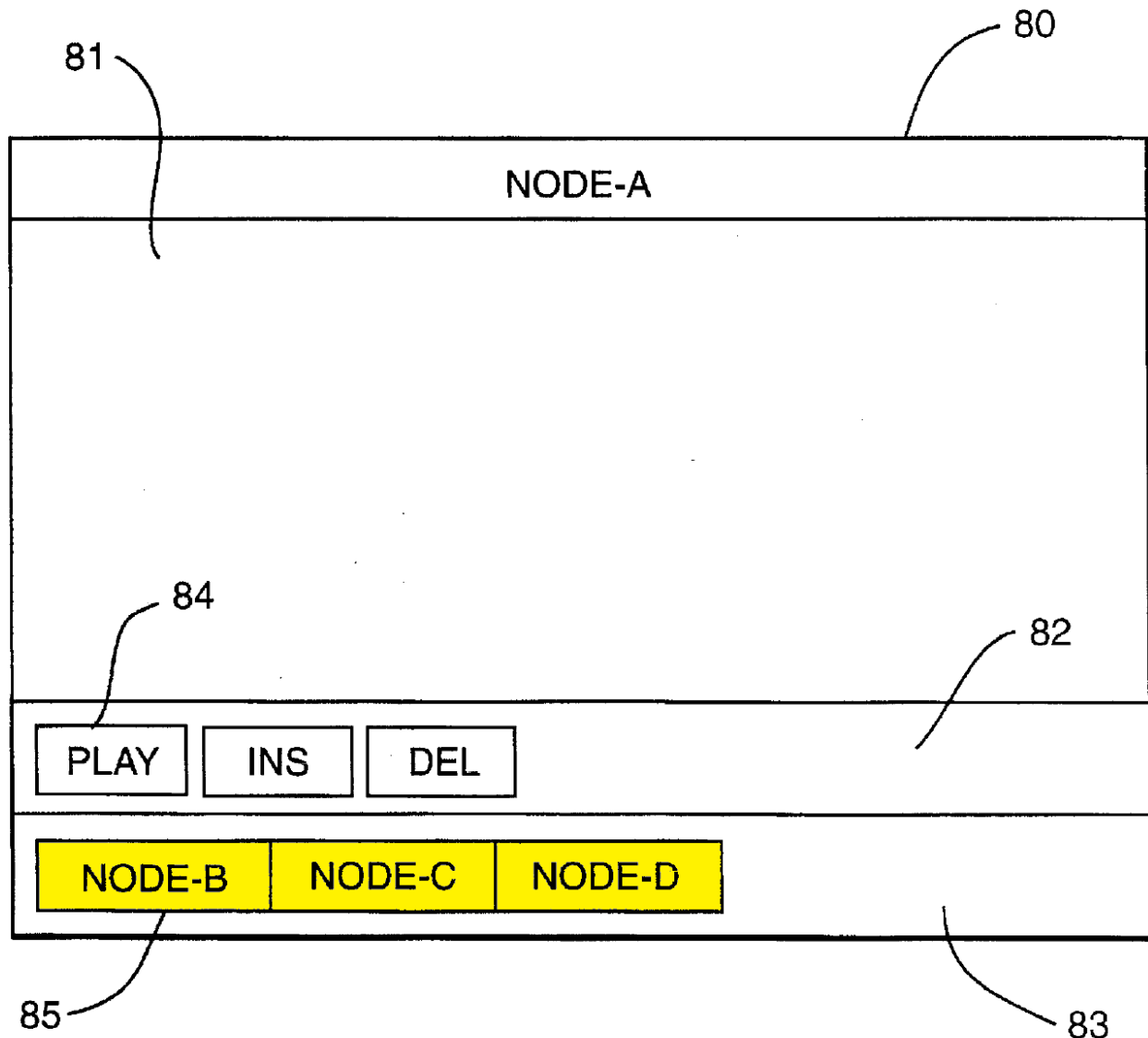


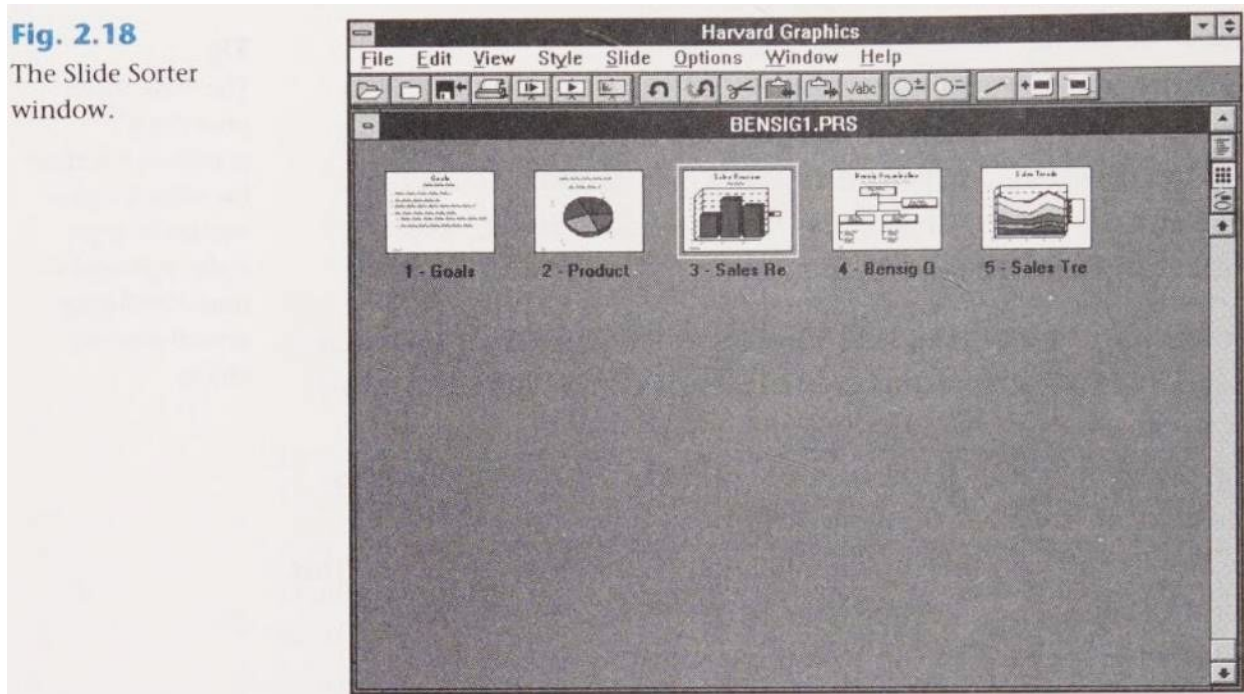
FIG. 8

(Shimizu, Fig. 8 (highlighting added), 7:12-14.) Figure 8 includes the display of a current node (NODE-A) in area 81, and in area 83, icons 85 representing each of the nodes linked starting from NODE-A. (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,**”

Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

because icons **85** identify each node from NODE-A and the order in which they appear in the show structure.

This functionality is also separately disclosed by a “Slide Sorter” feature in Benedict that displays an icon for each slide, numbered based on its order:



(Benedict, 0061, Fig. 2.18; *see also id.*, 0060 (“Figure 2.18 shows the Slide Sorter displaying five slides. The Slide Sorter represents these slides as scaled down images.”).) Based on Shimizu and Benedict, therefore, it would have been obvious to display the show structure to the user. (Schmandt, ¶¶163-164.)

11. Claim 11: “The 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 with a text box for the user to set or change the display time for one or more

slides. (Benedict, 00120, Fig. 17.1; *see also id.*, 00121 (“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. (Schmandt, ¶165.)

12. **Claim 14: “The 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.”**
13. **Claim 16: “The method in accordance with claim 14, wherein said control buttons provide for skipping of said content being presented.”**

These limitations are disclosed and rendered obvious by Benedict and Fox. 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, 00124.) Users can also define their own HyperShow links to jump to other parts of the presentation. (Benedict, 00125-00130.) These links take the form of buttons. (Benedict, 00127 (“You create HyperShow buttons in the Slide Editor.”).)

Fox also similarly discloses buttons for controlling the display of a series of web pages, including “the [Next] button [that] loads up the next page in the series, and the [Previous] button, naturally, loads up the previous page.” (Fox, 00262.)

It would therefore have been obvious to provide control buttons to enable the user to vary the duration content is 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 skip to a subsequent node without having to wait for expiration of the automatic advancement duration. (Schmandt, ¶¶167-168.)

14. Claim 17: “The method in accordance with claim 1, wherein said computer is a mobile device.”

This claim adds nothing of patentable significance. It was basic knowledge that the user computer (**“said computer”**) could have been a mobile device such as a laptop or mobile computer. Mobile devices were widely available by the early 1990s, several years before the priority date. (Schmandt, ¶¶169-170; *see also* EX1016, 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.”).) It would also have been obvious that a mobile device could have performed the functions of the user computer for claim 1. (Schmandt, ¶170.) An ordinarily skilled artisan would have been motivated to use a mobile device because such a device would have been portable and thus more convenient for the user. (*Id.*)

15. Claim 18: “The 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. (Schmandt, ¶110.) It would have been obvious to display that content through a web browser, the standard software tool for displaying content from the Web. (Schmandt, ¶171.)

16. Claim 19: “The 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.

Shimizu 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 that a user (using a mouse) can 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, 0075-0076.) It thus 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 alteration of the show structure. (Schmandt, ¶172.)

17. Claim 20: “The method in accordance with claim 1, wherein said show structure includes a sequential path of nodes.”

As fully explained for 1[a], the show structure under the proposed combination includes at least a sequential path of nodes. (Shimizu, *e.g.*, 7:62-65 (“...node A is displayed first followed by the contents of the nodes C, B and D, in that order...”), 9:1-4; *see also* Benedict, 0037 (“ScreenShows are on-screen presentations in which the program displays each slide in order.”).)

18. Claim 21: “The method in accordance with claim 1, wherein said show structure includes a concurrent path of nodes.”

Benedict and Fox also make clear that the nodes under the proposed combination need not be linked solely in single sequential fashion – there could be concurrent paths through them. For example, Benedict explains that slides can include hyperlinks to jump to other slides:

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.

(Benedict, 00116-00117.)

The capability to view nodes (which in Benedict are represented as slides) “in

any sequence” (*id.*) was basic 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. (Schmandt, ¶174.) For example, Fox discloses a web site five pages, each containing a hyperlink to the other four pages:

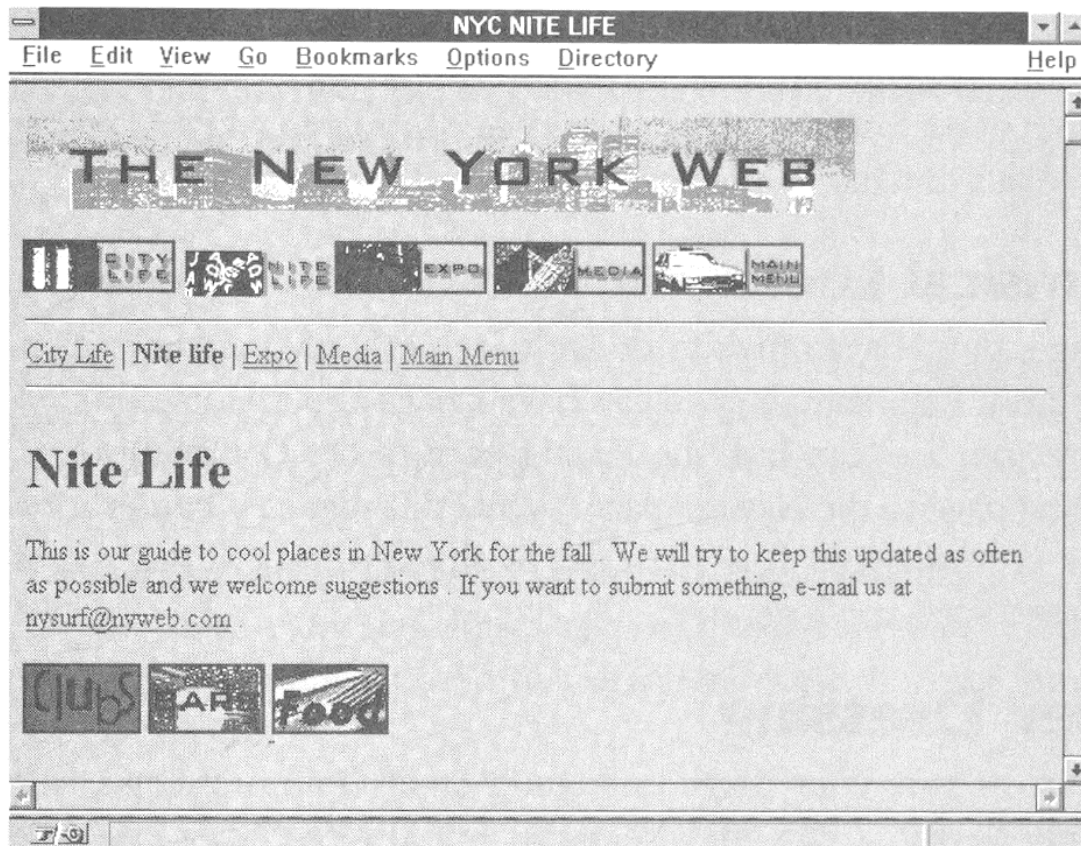
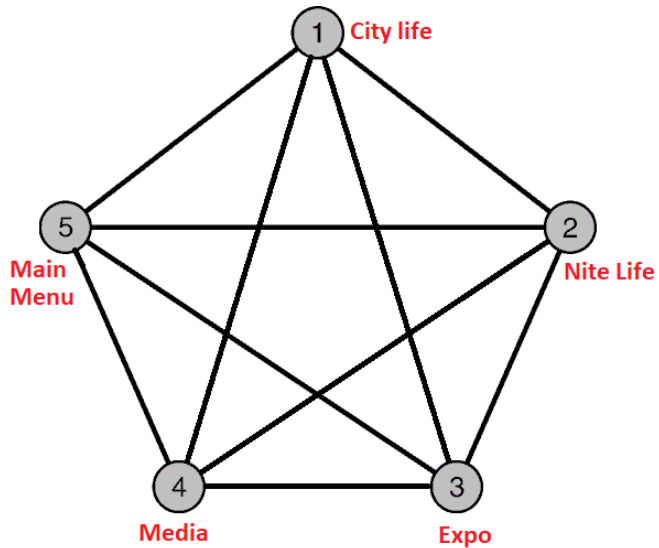


Figure 9-3 The New York Web, where each Web page contains links to other major sections

(Fox, 00263, Fig. 9-3.)

Figure 9-3 above shows an example page 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, 00262.) These links are also shown under the icons

as hyperlinks (as indicated by the underlining). 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 by clicking the hyperlink.



Each page thus links to each other page (at right). (Schmandt, ¶175.)

It therefore would have been obvious to implement the proposed combination such that the “**show structure of nodes**” for 1 [a] also includes “**a concurrent path of nodes,**” because simultaneous paths could have existed through the plurality of nodes. During operation of the system based on the proposed combination, therefore, 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 paths. (Schmandt, ¶176.)

Petitioner provided a full explanation of the motivation to combine Shimizu, Benedict and Fox for 1[a], which applies here. A skilled artisan would have been further motivated to provide a “**concurrent path of nodes.**” (Schmandt, ¶¶177-178.) 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, 00116-00117.) A concurrent path through the nodes have provided a useful enhancement to the proposed combination which would otherwise be limited to the sequential path established by the node and link information. (Schmandt, ¶178.)

C. Ground 2: Claims 63-67 Are Obvious Over Shimizu in View of Benedict, Fox, Wolfe, and Sagman

1. Independent Claim 63

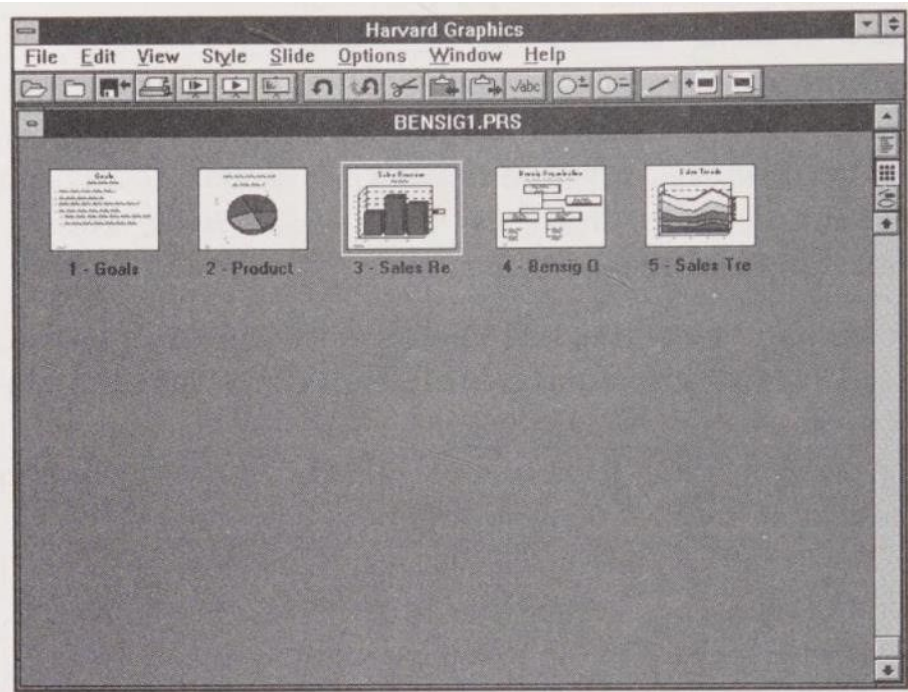
Claim 63 is similar in many respects to claim 1. The preamble recites, “[a] **method for presenting in an ordered arrangement on a user’s display, content derived via a communications network from a plurality of resources....**” As fully explained for claim 1, the Shimizu-Benedict-Fox combination renders 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 also presented “**in an ordered arrangement**” because, under the proposed combination as explained for claim 1, the content is presented in connection with the display of nodes presented in a specified order. (Schmandt, ¶¶103, 254.)

The content is also “**derived via a communications network from a plurality of resources.**” As explained, under the proposed combination with Fox as explained for claim 1, the content would have been 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 via the Internet (“**via a communications network**”). (Schmandt, ¶¶255, 102, 110-114.)

- (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;**” (63[a])

This limitation is disclosed by the combination of Benedict and Fox, both of which were discussed in connection with claim 1. With respect to Benedict, it discloses “**allowing the selection of at least one resource from a plurality of accessible resources.**” Benedict discloses a “Slide Sorter” feature that displays icons, one icon for each slide in a presentation:

Fig. 2.18
The Slide Sorter
window.



(Benedict, 0061, Fig. 2.18; *see also id.*, 0060 (“Figure 2.18 shows the Slide Sorter displaying five slides. The Slide Sorter represents these slides as scaled down images.”).)

Benedict confirms that each slide in a presentation can include at least one “**accessible resource**,” such as graphic and sound files. (Benedict, 00137 (“In Harvard Graphics, each slide in a presentation can have an associated sound... When the slide is displayed in a ScreenShow, Harvard Graphics plays the sound file for the viewers to listen to.”), 0098 (“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).)

Benedict also confirms that each resource is “**associated with a corresponding identifier displayed on a first section of said user’s display,**” which corresponds to an icon image corresponding to the slide that includes the resource as shown in Figure 2.18 above. Each icon provides an “**identifier**” because it visually identifies the slide for the user, and as explained below, can be separately 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. Figure 2.18 of Benedict above, for example, displays each icon in a particular area of the screen within a window showing slides for the current presentation.

Benedict 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, 0061 (“When you select a slide, a highlight box appears around the slide, indicating that it is active.”), 0072 (“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. (Schmandt, ¶258.)

Benedict does not appear to expressly disclose that resources are accessible “**via said 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. (Schmandt, ¶¶259, 108-114.) Fox confirms discloses that Web resources identified by URLs included the same types of resources that can be incorporated into slides in Benedict, such as graphic and sound files. (Fox, 0080 (“The Web combines graphics, video, sound, text, hypertext, and more.”), 0049 (“The World Wide Web gives every article, newsgroup, database, computer, file or other Internet fragment its own URL.”).) The motivation to combine to provide network-accessible resources, provided for 1[a], applies here. (Schmandt, ¶¶110-114.)

An ordinarily skilled artisan would also have been motivated to apply the 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. (Schmandt, ¶260.) Benedict provides an express motivation to combine by explaining that the Slide Sorter “enables you to view many slides simultaneously” (Benedict, 0060), thus allowing the user under the proposed combination to more

quickly review 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, 0071.)

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. (Schmandt, ¶¶261-262; Benedict, 0071-0072.) For example, the Slide Sorter interface in Benedict enables the user to easily reorder slides by simply selecting a slide and moving it to a new position. (Benedict, 0061, 0075-0076.) 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. (Schmandt, ¶262.) A skilled artisan would have had a reasonable expectation of success and could have implemented the proposed combination using conventional techniques. (Schmandt, ¶263.)

- (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;” (63[b])**

Turning first to the **“show structure of nodes,”** as explained for 1[a], the Shimizu-Fox-Benedict combination renders obvious **“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. 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. (Schmandt, ¶¶264, 110.) It also would have been obvious that the show structure was “**creat[ed]**,” as it must have been brought into existence to perform the node display and other functions described in Shimizu. (Schmandt, ¶105.)

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. (Schmandt, ¶265.)

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. (Benedict, 0061 (italics in original); *see also id.*, 0060 (“You can change the order of your slides by clicking and dragging a slide to a new location.”); *see also id.*, 00124 (“If you change the order of the slides, Harvard Graphics updates

the links automatically.”).) Shimizu discloses similar functionality for specifying the order of nodes using mouse operations through the graphical user interface. (Shimizu, 7:35-48.) Shimizu also discloses the ability to insert a new node at a particular position. (Shimizu, 9:5-30; Schmandt, ¶267.) Shimizu does not expressly state that drag-and-drop operations are used to perform these techniques, but this would have been obvious as shown below. (Schmandt, ¶¶267, 275-276.)

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.**” 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 show structure being created under the proposed combination, each node having one or more corresponding resources. A drag-and-drop feature would have predictably enabled “**generation of a list of said resources by dragging and dropping said identifiers,**” because drag-and-drop operations on the icons (“**said identifiers**”) would have been used to arrange the nodes in the desired sequence. (Schmandt, ¶268.) Each node, as explained, is associated with one or more resources.

Claim 63 does not specify any particular format or organization for the claimed “**list.**” A skilled artisan would have appreciated that “**list**” does not require a single data structure stored contiguously in memory. For example, a “linked list”

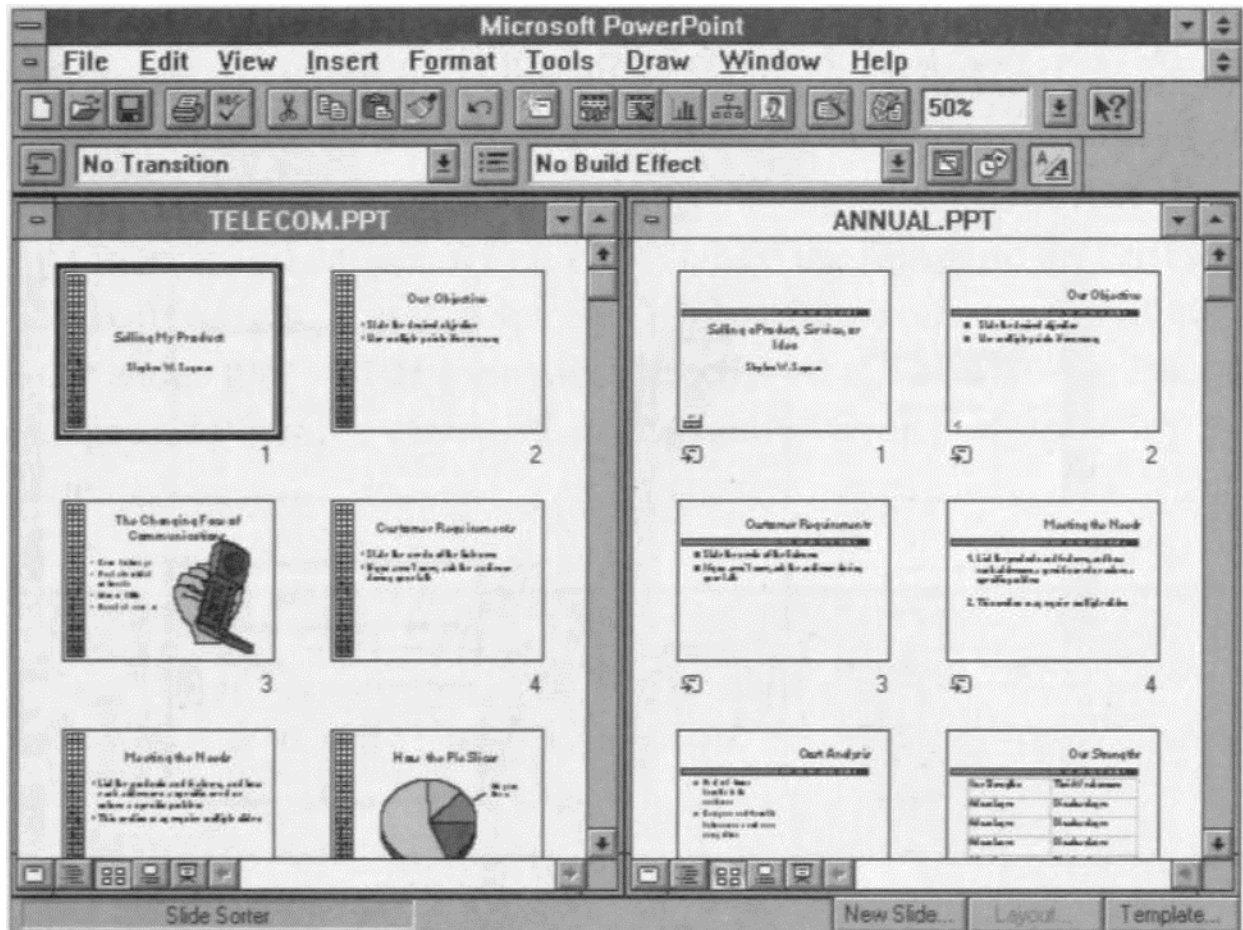
is a well-known data structure in which each node in the list may be stored in different locations and is connected to other nodes via links or pointers. (Schmandt, ¶269 (citing *Microsoft Press Computer Dictionary* (2d ed. 1994), EX1020, 008 (“**linked list** In programming, a list of nodes or elements of a data structure connected by pointers.”))).) This is precisely what the proposed combination discloses. 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 node information of the next node in the show structure with which it is connected. (Schmandt, ¶269; Shimizu, 1:61-63, 2:5-9.) The proposed combination therefore renders obvious generating “**a list of said resources.**”

Patent Owner may argue that Benedict does not render obvious “**dragging and dropping said icons on a second section of said user’s terminal,**” based on the argument that icons in the Slide Sorter in Benedict appear within a single graphical window. But the claim merely requires a first and second “**section,**” which does not appear to require physically separated areas. Benedict allows the user to drag an icon from its original location (“**a first section**”) to another position (“**a second section**”) on the user’s terminal. (Schmandt, ¶270.)

Nevertheless, even if the claim were interpreted to require two physically separated sections of the screen, this would have been obvious in further view of Sagman, which describes functionality of Microsoft PowerPoint. (Sagman, 0028-

Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

0029; Schmandt, ¶¶85-86, 271.) Sagman discloses a “Slide Sorter” similar to Benedict but with an additional capability – the ability to display two presentations in separate windows such that slides from one presentation can be moved or copied into another using drag-and-drop techniques:



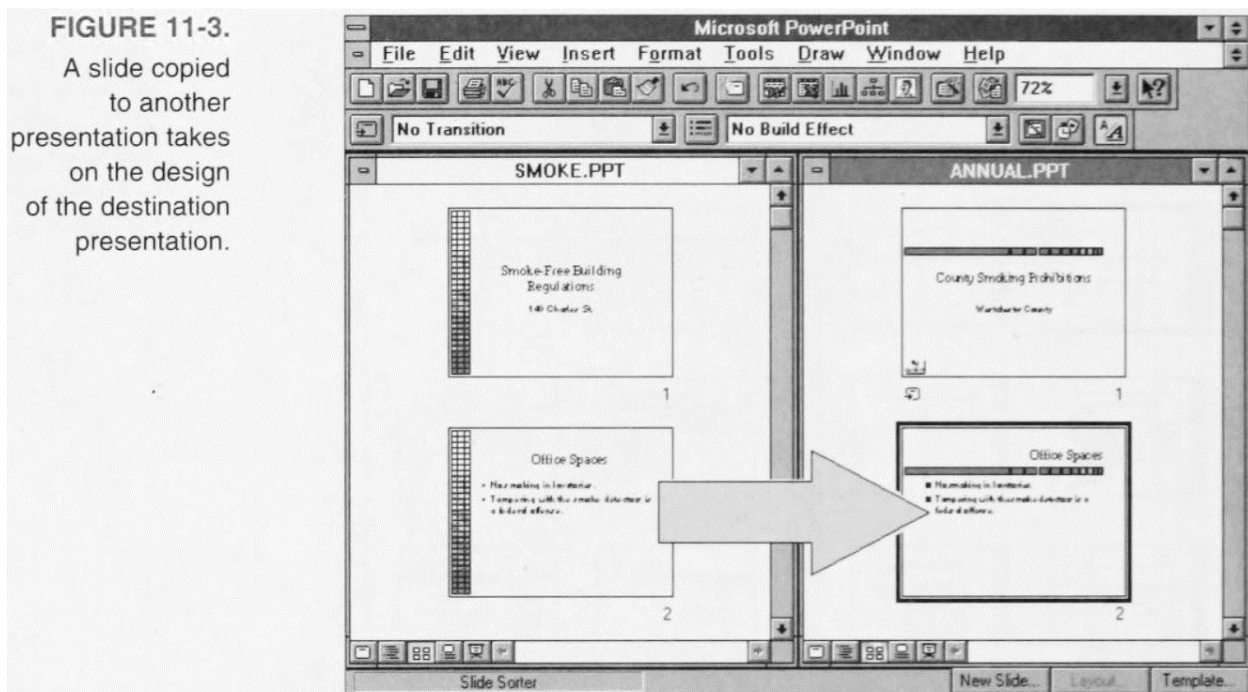
(Sagman, 0056.)

The screen display above shows two PowerPoint presentations (TELECOM.PPT and ANNUAL.PPT) in separate side-by-side windows, each having icons for individual slides similar to Benedict. Sagman explains that, using standard drag-and-drop techniques, the user can copy a slide from the first

presentation into the second using drag-and-drop:

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.

(Sagman, 0058.) Figure 11-3 shows an example of copying a slide from a first presentation into a second:



(Sagman, Fig. 11-3, 0057.) In this example, the slide “Office Spaces” was copied from SMOKE.PPT to the second position in ANNUAL.PPT. As shown, the copied slide retains the same content as the original slide but takes on the design of the target presentation. (Schmandt, ¶271.)

Rationale and Motivation to Combine (Shimizu, Benedict and Fox with Sagman): It would have been obvious to combine the Shimizu-Benedict-Fox

system with Sagman, predictably resulting in the proposed combination 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 one window (in **“a first section of said user’s display”**) into the show structure being created in another (to **“a second section of said user’s display”**). The drag-and-drop operation thus **“generat[es] a list of said resources”** because it specifies a sequence of nodes for the show structure being created, each node identifying the one or more resources. (Schmandt, ¶272.) Under the proposed combination with Shimizu, as explained, each resource would have been identified by URLs and recorded in the node information of its corresponding node as explained. (Schmandt, ¶¶272, 110; Shimizu, 5:55-59.) Like Benedict discussed above, Sagman is also an analogous reference in the field of providing organized or sequenced presentation of content, and would have been reasonably pertinent to a number of problems facing the inventor. (Schmandt, ¶273.)

An ordinarily skilled artisan would have been motivated to combine in order to improve user experience. (Schmandt, ¶274.) Sagman provides an express motivation to combine by explaining that using drag-and-drop across two presentations “lets you transfer an existing slide to a new presentation rather than recreate its content.” (Sagman, 0057.) This would have enabled the user to more

quickly create new nodes and specify their sequence by moving or copying existing nodes to their desired locations in the new show structure, beneficially obviating the need to create them from scratch. (Schmandt, ¶274.)

Like Sagman, Benedict discloses the ability to move or copy slides from one presentation to another but instead of drag-and-drop, using on-screen menus. (Benedict, 0076-0079.) An ordinarily skilled artisan would have been motivated to improve Benedict by using drag-and-drop techniques as described in Sagman, which were standard features of conventional user interfaces and had well-known advantages. Drag-and-drop techniques had advantages over on-screen menus because they allowed users to move on-screen objects using gestures that mimic how human beings manipulate objects in the real world, making it easy to learn and use and a particularly desirable choice for user interface design. (Schmandt, ¶¶275-276.) 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. (Schmandt, ¶277.)

- (c) **“without requiring user input, automatically accessing...” (63[c])**
- (d) **“automatically retrieving content...” (63[d])**
- (e) **“causing the presentation of said retrieved content....” (63[e])**

These limitations are substantially similar to 1[b]-1[c] above. Under the

proposed combination as explained, each resource identified by each of the nodes would have been automatically accessed, retrieved, and displayed in sequential fashion, without requiring user input. (Schmandt, ¶¶278-279, 129-148.) Based on the further combination with Wolfe, as explained for 1[c], it would have been obvious that during presentation content from the accessed resources, at least some of the other network accessible resources are being accessed. (Schmandt, ¶280.)

2. **Claim 64: “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.”**

As fully explained for 1[d], it would have been obvious to provide the claimed interactively variable duration information. (Schmandt, ¶¶149-155, 122-128.)

3. **Claim 65: “The 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.”**

Claim 65 would have been obvious based on disclosures in Shimizu and Benedict. Starting with Shimizu, it discloses the ability to display icons **85** in Figure 8 at the same time as node content in display area **81**.

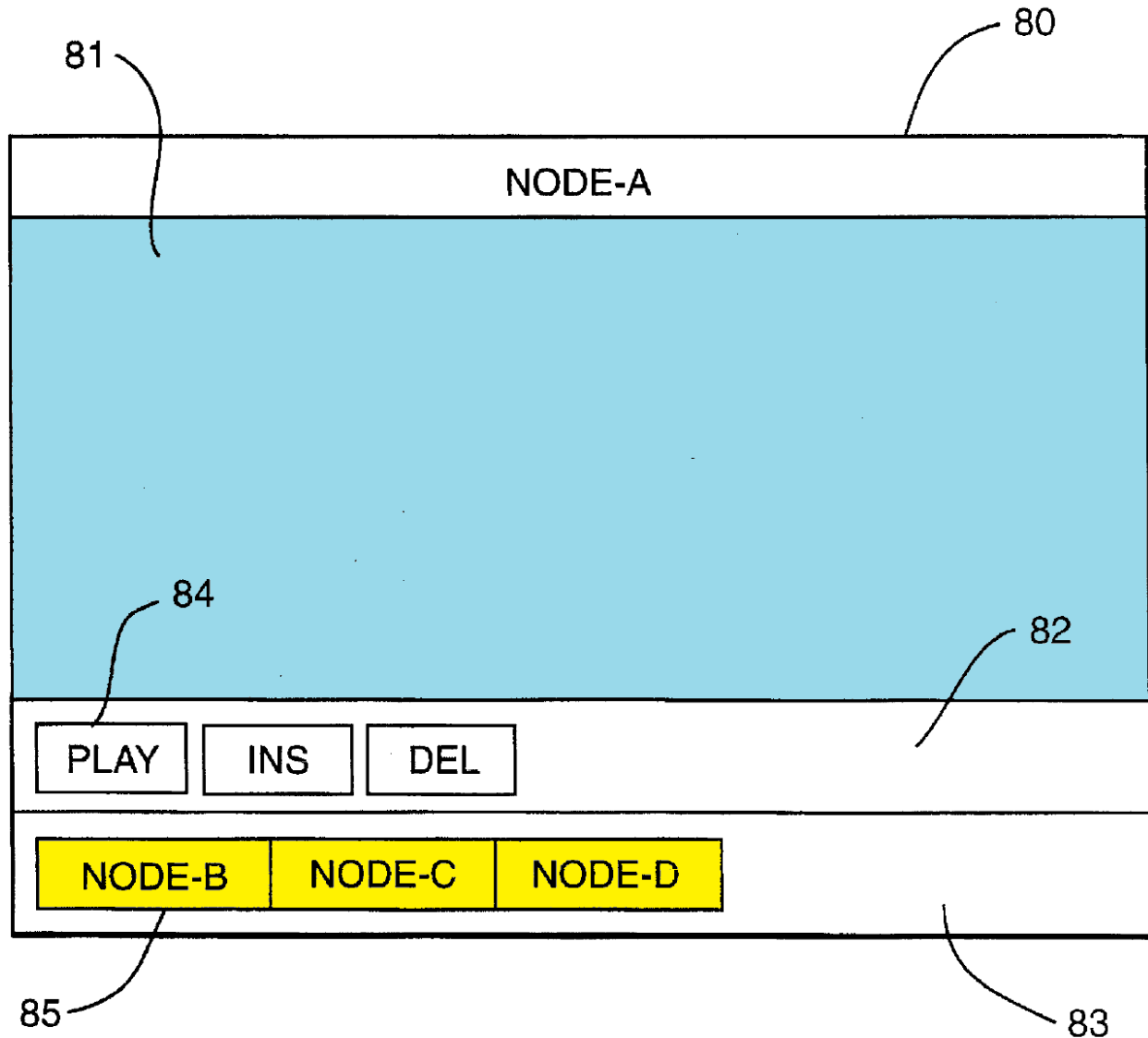


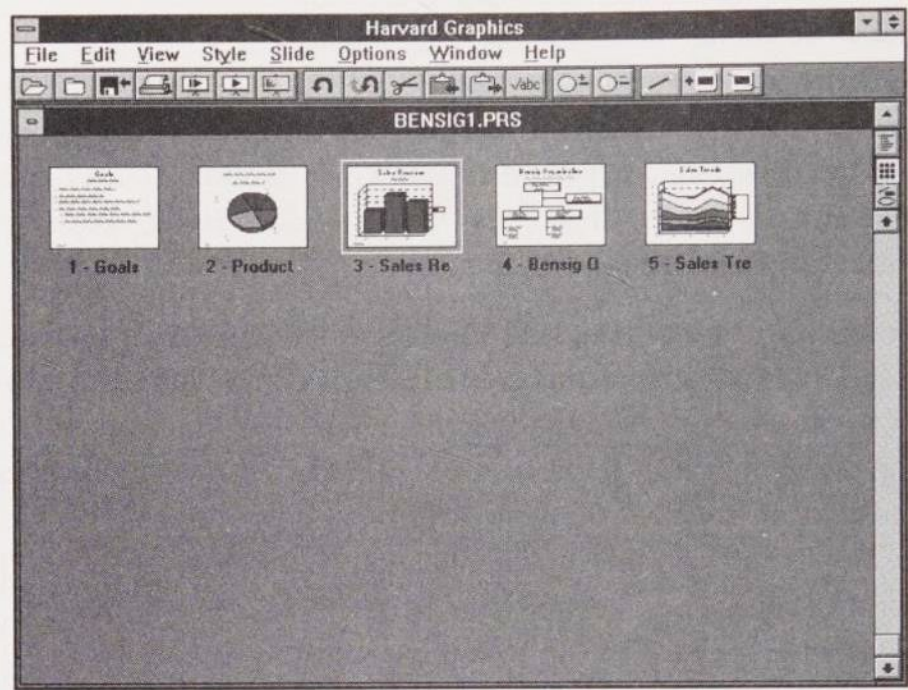
FIG. 8

(Shimizu, Fig. 8 (highlighting added).) Display area 81 for presents the content of NODE-A, the currently displayed node. Icons 85 show icons for the other three nodes in the show structure. (Shimizu, 7:16-18, 7:20-24, 7:62-65, 7:52-59.)

As explained for 63[a], like Shimizu, Benedict displays icons but also shows smaller versions of the contents of their corresponding slides:

Fig. 2.18

The Slide Sorter
window.



(Benedict, 0061, Fig. 2.18; *see also id.*, 0060.)

It would have obvious to combine these user interface elements from Shimizu and Benedict, predictably resulting in each node icon **85** being presented with its underlying content, such as a smaller version disclosed in Benedict. (Benedict, 0071 (“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 Petitioner’s expert created for illustration shows example of this combination:

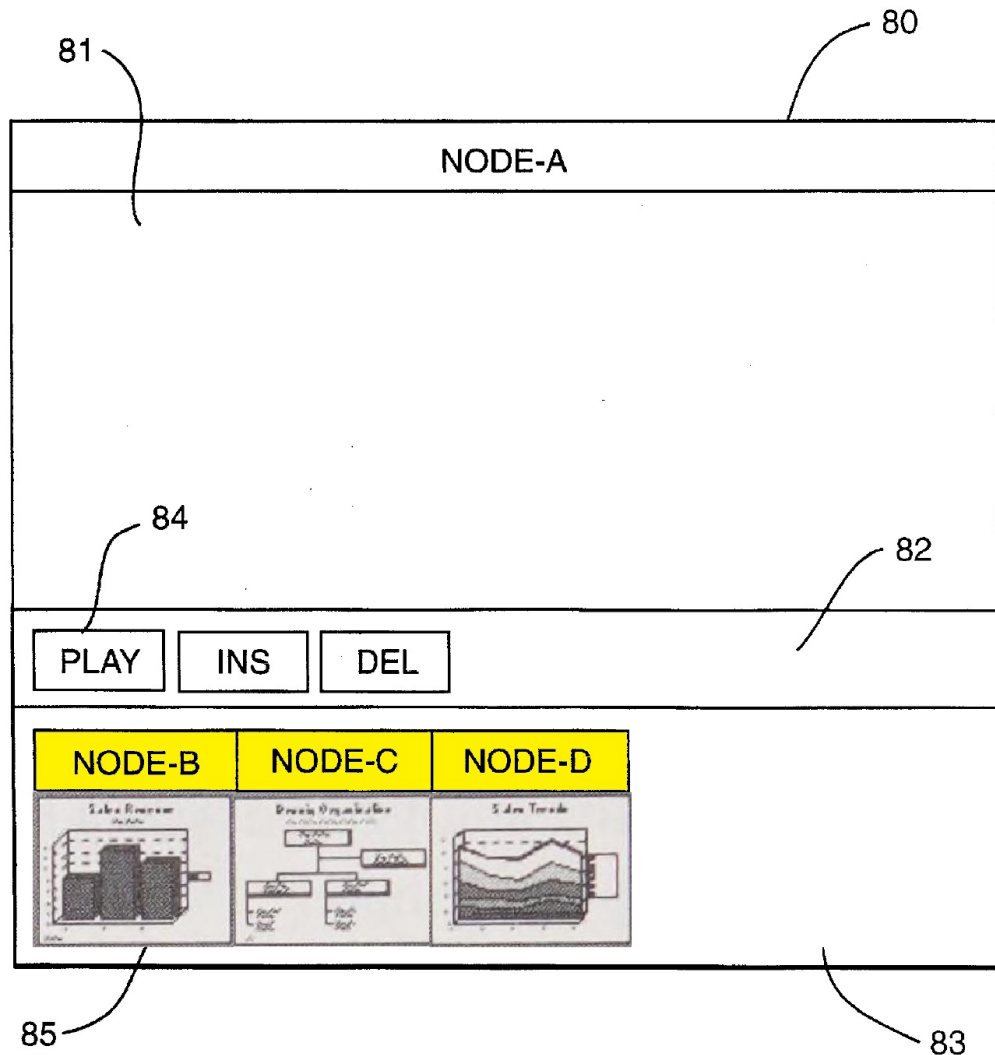


FIG. 8

(Schmandt, ¶286.) Each node icon **85** above, as shown, shows its underlying content (in smaller format), consistent with 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.”** In this example, during presentation of the content of NODE-A, the user can also see content from three other nodes (NODE-B, NODE-C, NODE-D).

A 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. (Schmandt, ¶288.) The icon 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). (*Id.*; Benedict, 0071-0072.) Petitioner has also provided separate **Ground 4** below showing why claim 65 would also have been obvious in view of Robertson.

4. Claim 66: “The 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.”

It would have been apparent and obvious that the accessing step of 1[b] would have required executing an application file on the user’s computer, *e.g.*, a software program used to access and display content associated with the nodes. (Schmandt, ¶¶289-290; Shimizu, *e.g.*, 9:63-66, 5:40-42; *see also* Benedict, 00149.) Under the further combination with Fox, a web browser on the user’s computer would have displayed content resources from the Web. In any case, it would have been apparent and obvious that the software used to access and display resources under the proposed combination would have included at least an application file (*e.g.* executable or “.EXE” file) executing on the user’s computer. (Schmandt, ¶291.)

5. Claim 67: “The method in accordance with claim 66 further comprising the step of generating a data file resulting from said executing step.”

It is unclear whether “**generating**” a data file requires that a new data file be created or encompasses creating a copy of an existing data file. With respect to the latter interpretation, a data file containing the content resource associated with a node would have been downloaded over the Internet from a remote server in response to a request by the user’s computer (*e.g.* a web browser), then stored on the user’s computer. (Schmandt, ¶293.) It thus would have been obvious that a data file would have been generated on the user computer as a result of the executing step. (Schmandt, ¶293.) This was also built-in functionality of “cache” and “history” features of conventional web browsers. (Schmandt, ¶293; Fox, 00113 (“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.”), 00167 (same; Netscape browser), 00192.)

And if “**generating**” were interpreted to require creation of a new data file, this would also have been obvious. It was well-known that web servers created new content in response to requests to access resources. Fox describes a well-known process known as the Common Gateway Interface (CGI) in which web servers generated Web content on-the-fly in response to requests from browsers. (Fox, 00382, 00411 (“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 CGI script that generates a new image for a web page showing an image currently captured by a camera. (Fox, 00412, Fig. 15-2.) It was also well-known to use CGI techniques to generate HTML on-the-fly in response to a user request, as discussed in Fox. (Fox, 00409-00411.) It would have been obvious to generate new content for a wide variety of purposes, such as financial, weather, or other topics where information changes. (Schmandt, ¶295.)

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 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. (Schmandt, ¶¶294, 296.) 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), the server identified in the URL would have responded to the request from the user computer by creating a new data file. (Schmandt, ¶296.) 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, 00411.) It would also have been obvious that these resources would have taken the form of “**a data file**,” because

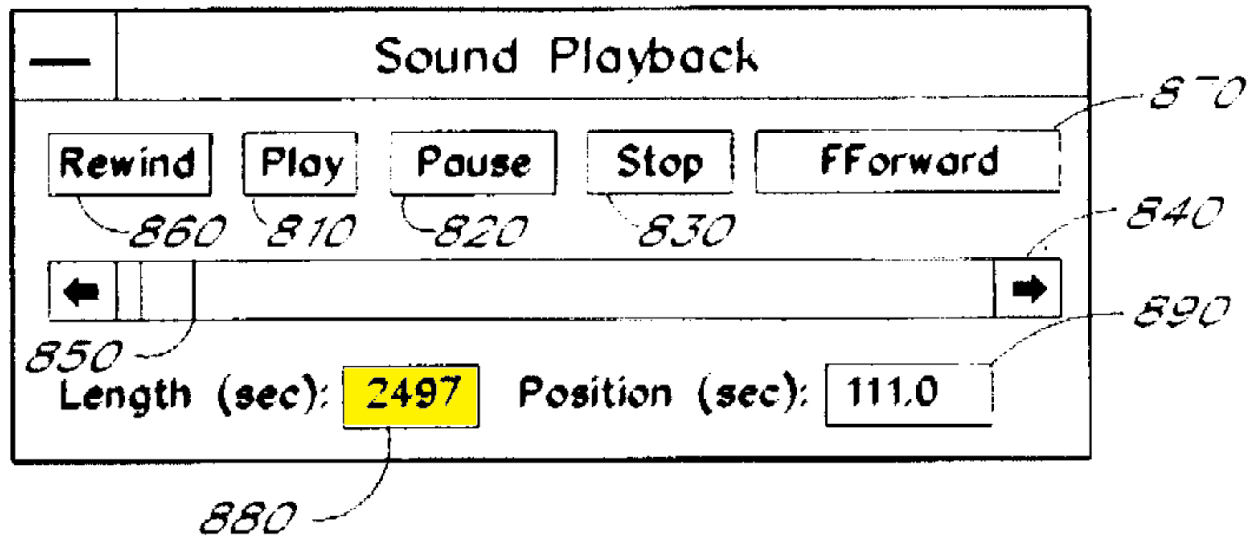
data files are a fundamental unit of storage in a computer system and used by web servers and browsers to store Web resources. (Schmandt, ¶298.)

D. Ground 3: Claims 11-16 Are Obvious Over Ground 1 Prior Art With Glaser

- 1. Claim 11: “The method in accordance with claim 1, wherein said interactively variable duration information is displayed to the user.”**
- 2. Claim 12: “The method in accordance with claim 11, wherein said display of said interactively variable duration information to the user occurs during presentation of said content.”**

As explained for **Ground 1** (claim 11), Benedict that the interactively variable duration information is displayed to the user. (Schmandt, ¶165; Benedict, 00120-00121, Fig. 17.1.) 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. Petitioner has therefore cited **Glaser** in combination with **Ground 1** for claims 11-12.

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.) Petitioner cites Glaser for a narrow purpose of disclosure of user interface features relating to the playback of 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, 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.)

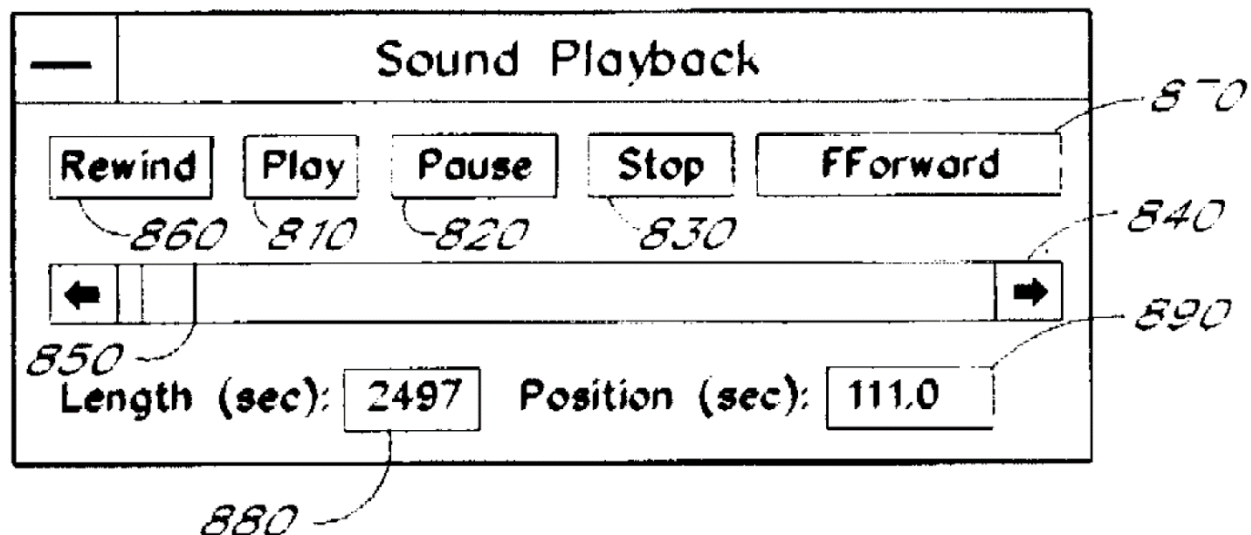
Rationale and Motivation to Combine (Ground 1 Prior Art with Glaser): It would have been obvious to combine the prior art for **Ground 1** with Glaser, predictably resulting in the proposed combination with the further ability to display the “interactively variable duration information” as described for claim 1[d], “during presentation of said content,” for example via length field 880 above. In operation, therefore, the system would have displayed how long the content

corresponding to the resource will be presented before automatically advancing to the next node. (Schmandt, ¶303.) Although not required by claims 11-12, it would also have been obvious to further display how long the current content has been presented such as provided by field **890**. (*Id.*) 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. (Schmandt, ¶304.)

A skilled artisan would have appreciated that this feature would have provided useful information to keep the user's attention. For example, in the case of visually-presented content, the user may have reviewed the content quickly and is thus awaiting the automatic advancement to the next node. Without display of the duration information (for example length field **880** of Glaser), that user may not know how long it must wait and thus might shift its attention elsewhere, potentially resulting in missing content. (Schmandt, ¶305.) 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 know when the next slide will appear so it can use its time efficiently and pay attention when needed. (*Id.*) A skilled artisan would have had at least a reasonable expectation of success and could have easily implemented the proposed combination. (Schmandt, ¶306.)

3. **Claim 13:** “The method in accordance with claim 1, further comprising the step of providing a play bar to enable the user to vary said duration that said content is being presented.”
4. **Claim 14:** “The 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.”
5. **Claim 15:** “The method in accordance with claim 14, wherein said control buttons provide for pause, fast forward and rewind functions to be applied to said content being presented.”
6. **Claim 16:** “The method in accordance with claim 14, wherein said control buttons provide for skipping of said content being presented.”

Claims 13-16 are closely-related and can be addressed together. 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).)

Figure 8 above 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, for example, 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** or fast forward **870** skip to a later location or the end of the clip.

Petitioner also notes that Glaser’s user controls are not limited to controlling 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.)

A 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 presentation of content. (Schmandt, ¶310.) Benedict confirms that each of the slides in a presentation can include sound files. (Benedict, 00137.) 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 as claimed. (Schmandt, ¶310.)

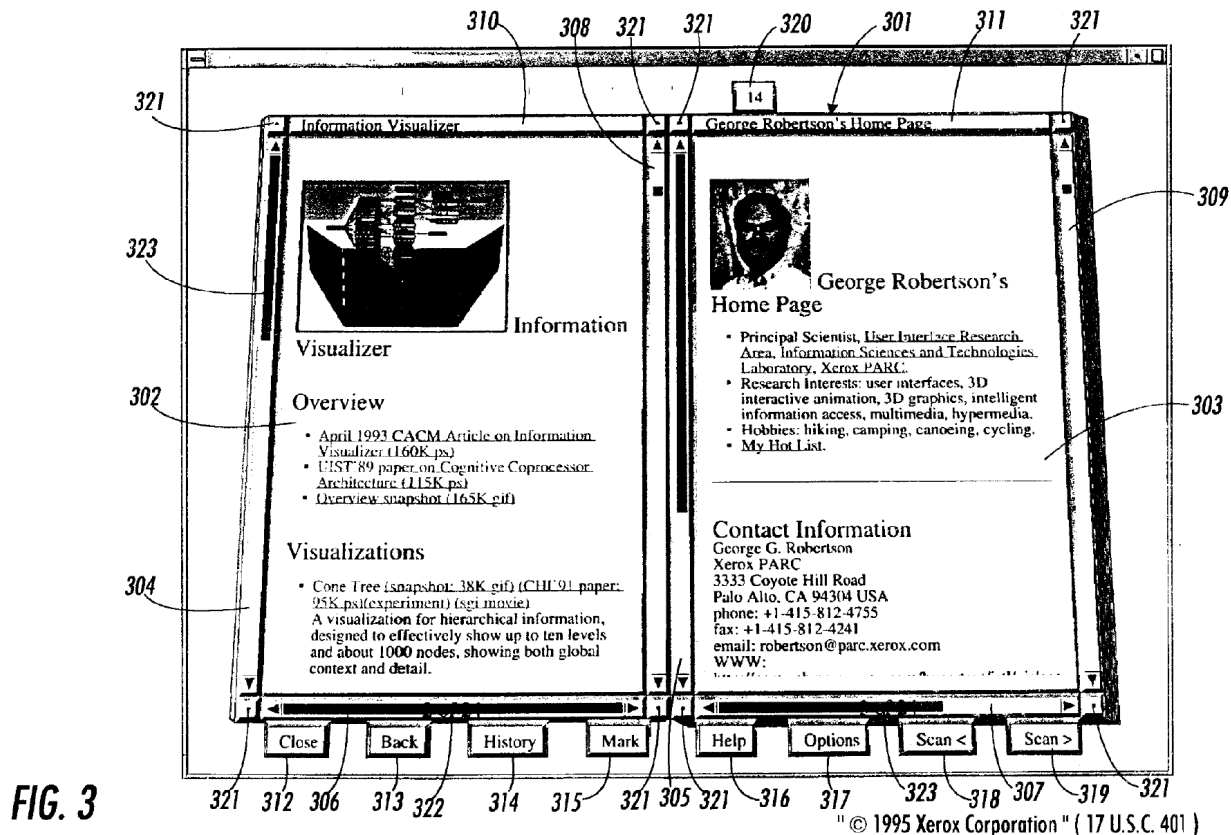
The features shown in Figure 8A, such as pausing, fast forwarding and rewinding would have been widely familiar and to ordinarily skilled artisans and laypersons, as they were derived controls that existed since at least the 1970s with tape players. (Schmandt, ¶311.) A skilled artisan would have been motivated to provide a play bar with these control buttons for the same reasons tape players have long-included 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. (*Id.*)

E. Ground 4: Claim 65 is Obvious Over Ground 2 Prior Art With Robertson

As noted, claim 65 depends from claim 63 and further recites “**the presentation of content from at least two nodes presented simultaneously.**” Claim 65 is obvious for the reasons provided for **Ground 2**. But as an additional basis, this claim would have been obvious in further view of **Robertson**.

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.) Robertson presents a list of web pages by displaying each web page in the list as a “page” of a virtual book (WebBook). (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.) Figure 3 shows an exemplary WebBook view showing two web pages visible simultaneously:



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

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, 6:16-19; *see also id.*, 6:28-30.) “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.) Another embodiment 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.) Robertson thus discloses presenting content from at least two pages simultaneously.

Rationale and Motivation to Combine (Ground 2 Prior Art with Robertson):

It would have been obvious to combine the references cited for **Ground 2** with Robertson. The combination would have predictably yielded the system based on the proposed combination of **Ground 2** in which the automatic presentation of content of claim 63 “**includes the presentation of content from at least two nodes presented simultaneously.**” For example, during automatic presentation of the retrieved content for the current node, the content from the next sequential node could also have been presented at the same time. (Schmandt, ¶324.) Robertson is an analogous reference in the same field and would have been reasonably pertinent to a number of problems facing the inventor. (Schmandt, ¶325.)

A skilled artisan would have been motivated to combine in the manner described above to improve user experience. (Schmandt, ¶326.) Robertson explains that organizing information using a “book metaphor” was a known way to present information. (Robertson, 1:66-2:3.) Displaying multiple nodes at the same time using a “book metaphor” would have advantageously leveraged the user’s familiarity with real-world books. And displaying two or more nodes at the same time would also have more efficiently utilized screen space and improved the user’s ability to review and fully visualize node content. (Schmandt, ¶326.) A skilled artisan would have had a reasonable expectation of success and could have implemented the feature of claim 65, and the proposed combination, using

Petition for *Inter Partes* Review of
U.S. Patent No. 8,965,932 B2

conventional programming techniques. (Schmandt, ¶327.)

VII. CONCLUSION

Petitioner respectfully requests IPR institution.

Dated: February 21, 2023

Respectfully submitted,

COOLEY LLP
ATTN: Patent Group
1299 Pennsylvania Avenue NW
Suite 700
Washington, DC 20004
Tel: (650) 843-5001
Fax: (650) 849-7400

By: / Heidi L. Keefe /
Heidi L. Keefe
Reg. No. 40,673
Counsel for Petitioner

CERTIFICATE OF COMPLIANCE WITH WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), I certify that this petition complies with the type-volume limits of 37 C.F.R. § 42.24(a)(1)(i) because it contains 13,964 words, according to the word-processing system used to prepare this petition, excluding the parts of this petition that are exempted by 37 C.F.R. § 42.24(a) (including the table of contents, a table of authorities, mandatory notices, a certificate of service or this certificate word count, appendix of exhibits, and claim listings).

DATED: February 21, 2023

COOLEY LLP
ATTN: Patent Docketing
1299 Pennsylvania Avenue NW
Suite 700
Washington, D.C. 20004
Tel: (650) 843-5001
Fax: (650) 849-7400

/ Heidi L. Keefe /
Heidi L. Keefe
Reg. No. 40,673

CERTIFICATE OF SERVICE

I hereby certify, pursuant to 37 C.F.R. Sections 42.6 and 42.105, that a complete copy of the attached **PETITION FOR INTER PARTES REVIEW OF U.S. PATENT NO. 8,965,932 B2**, including all exhibits (**Nos. 1001-1010, 1012-1013, 1016-1017, 1019-1031**) and related documents, are being served via Federal Express on the 21st day of February, 2023, the same day as the filing of the above-identified document in the United States Patent and Trademark Office/Patent Trial and Appeal Board, upon Patent Owner by serving the correspondence address of record with the USPTO as follows:

135621 - IPSILON USA
110 W 40th Street
Suite 2001
New York, NY 10018

And, via Federal Express upon counsel of record for Patent Owner in the litigation pending before the U.S. District Court for the District of Delaware entitled *Robocast, Inc. v. YouTube, LLC*, Case No. 1-22-cv-00304 (D. Del.) as follows:

Andrew C. Ryan
Cantor Colburn LLP
20 Church Street
22nd Floor
Hartford, CT 06103
(860) 286-2929, ext. 1127
aryan@cantorcolburn.com

DATED: February 21, 2023

/ Heidi L. Keefe /
Heidi L. Keefe
Reg. No. 40,673
COOLEY LLP
1299 Pennsylvania Ave. NW,
Suite 700
Washington, D.C. 20004
Tel: (650) 843-5001