

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION

Petitioner

v.

CYPRESS LAKE SOFTWARE, INC.

Patent Owner.

PTAB Case No.: To Be Assigned
Patent No. 9,423,954

**DECLARATION OF DR. LOREN TERVEEN
IN SUPPORT OF PETITION FOR POST-GRANT REVIEW
OF U.S. PATENT NO. 9,423,954**

TABLE OF CONTENTS

I.	Introduction.....	1
II.	Summary of Opinions	1
	A. Education and Experience.....	5
	B. Compensation	7
	C. Documents and Other Materials Relied Upon	7
III.	Statement of Legal Principles	7
	A. Claim Construction	7
	B. Written Description.....	8
	C. Benefit of Claimed Priority / Effective Filing Date.....	9
	D. Anticipation.....	9
	E. Obviousness	10
IV.	Overview of the '954 Patent	10
	A. Summary of the '954 Patent	10
	B. Prosecution History.....	21
	C. Related District Court Litigation	22
V.	State of the Art Prior to the '954 Patent.....	22
	A. Correlated User Interface Elements	22
	B. Window Layout via “Snapping”	28
	C. The Level of Ordinary Skill in the Art.....	31
VI.	Identification of the Prior Art and Summary of Opinions	33
VII.	Unpatentability of the Challenged Claims of the '954 Patent	34
	A. Ground 1: Claims 1-20 Are Invalid Under 35 U.S.C. § 112(a) For Lack Of Written Description.....	34
	1. Independent Claims 1, 14, and 19.....	34
	a. No Written Description Support for Presenting a Second Window Adjacent to a First Window In Response To User Input	36
	2. Dependent claims 2-13, 15-18, and 20	45
	B. Ground 2: Claims 1-2, 4-6, 8, 10-15, and 16-20 Are Invalid Under 35 U.S.C. § 102(a)(1) as Anticipated By The Public Use and Sale of Microsoft’s Windows 10 Operating System.....	46
	1. Independent Claim 1	47

2.	Dependent Claims 2, 4, 5, 6, 8, 10, and 11	52
3.	Dependent Claims 12 and 13	57
4.	Independent Claim 14	62
5.	Dependent Claims 16, 17, and 18	65
6.	Claims 19 and 20	68

I. INTRODUCTION

1. My name is Loren G. Terveen. I am currently a full Professor of Computer Science & Engineering at The University of Minnesota, and hold the title of Distinguished McKnight University Professor.

2. I have been engaged by Microsoft Corporation (“Petitioner”) in connection with the Petition for Post-Grant Review (“PGR Petition”) of U.S. Patent No. 9,423,954 (the “’954 patent”).

3. I understand that the ’954 patent has been assigned to Cypress Lake Software, Inc. (“Cypress Lake”).

4. This declaration is based on the information currently available to me. To the extent that additional information becomes available, I reserve the right to continue my investigation and study, which may include a review of documents and information that may be produced, as well as testimony from depositions that have not yet been taken.

II. SUMMARY OF OPINIONS

5. It is my opinion that independent claims 1, 14, and 19 of the ’954 patent are invalid for lack of written description support under AIA 35 U.S.C. § 112, as explained to me by Petitioners’ counsel.

6. The ’954 patent describes methods and systems for identifying programmable “binding information” that defines mappings between arbitrary user

interface components across different applications. Ex. 1001 at 12:56-62 & Fig. 2. Those methods and systems then enforce the mappings identified in the binding information by detecting changes in user interface components and making corresponding changes in other components according to those pre-identified mappings. *Id.* at 16:6-16, 17:28-43 & Fig. 2. The result is a flexible system that allows a programmer to arbitrarily define, e.g., using XML or custom source code, how certain user interface elements should respond to changes in other user interface elements. *Id.* at 13:21-15:7 & Fig 7. Once the binding information has been specified by the programmer and identified by the system, the system automatically enforces those bindings by monitoring all changes to the user interface elements and orchestrating corresponding changes in other user interface elements as required by the pre-identified mappings. *Id.* at 16:6-16, 17:28-43 & Figs. 2 & 3. Every disclosed embodiment requires this encoded binding or mapping information — it is the central premise of the entire patent.

7. The claims of the '954 patent are entirely divorced from the concept of “binding information,” despite this being the central premise of the invention described in the specification. The claims do not mention “binding information” or pre-identified “mappings” between user interface elements. There are no claim requirements that binding information be identified, and no description of using identified binding information to implement mappings between user interface

components. Each of the claims are directed to sequences of user inputs that result in some change in the user interface, with no reference or relationship to the methods and systems actually described in the specification.

8. Specifically, each of the independent claims of the '954 patent requires “detection of a [second/third] user input” and “in response to [that] user input” presenting a “second window” adjacent to an already presented “first window.” *Id.*, claims 1, 14, 19. The specification never describes presenting a second window adjacent to a first window based on a detected “user input.” The specification only identifies detecting user inputs in two contexts: (1) identifying binding information, *id.* at 15:53-65, and (2) detecting changes to a user interface element that is part of a mapping in pre-identified binding information, *id.* at 17:7-8. Neither of those disclosures in the specification correspond to what is claimed.

9. A person of ordinary skill in the art (“POSITA”) reading the specification would not have understood the inventors to have been in possession of an invention consisting of “detecting a [second/third] user input” and presenting a “second window” adjacent to an already presented “first window” in response to that user input without the use of pre-identified binding information. Therefore, it is my opinion that claims 1, 14, and 19 are invalid for lack of written description support.

10. It is my opinion that the dependent claims 2-13, 15-18, and 20 are also

invalid for lack of written description support under AIA 35 U.S.C. § 112. Each of those claims depends on either claim 1, 14, or 19. None of those dependent claims require “binding information” or otherwise rectify the lack of written description support in the independent claims.

11. Finally, it is my opinion that if the claims of the '954 patent are not entitled to an effective filing date earlier than the actual filing date of October 27, 2015, claims 1-2, 4-6, 8, 10-15, and 16-20 are rendered obvious by the prior use and sale of Windows 10 under Cypress Lake's own interpretation of the claims. I understand that Cypress Lake has filed an action in district court and alleged that devices running Windows 10 infringe these claims of the '954 patent. I have reviewed the claim charts that Cypress Lake has provided in that litigation, which allege that devices running Windows 10 satisfy each and every of one of the limitations in claims 1-2, 4-6, 8, 10-15, and 16-20. I further understand that the Windows 10 functionality that Cypress Lake alleges to infringe was in public use and/or publically known at the time of the invention. Because the use that Cypress Lake alleges to infringe actually pre-dates the effective filing date of the '954 patent, the asserted claims of the '954 patent would be rendered obvious under Cypress Lake's interpretation of the claims.

12. The subsequent sections of this declaration will first provide my qualifications and experience and then describe the details of my analysis and

observations regarding the '954 patent.

A. Education and Experience

13. I received a B.A. in Computer Science, Mathematics, and History from the University of South Dakota in 1984, a M.S. in Computer Science from the University of Texas in 1988, and a Ph.D. from the University of Texas in Computer Science in 1991.

14. I am a member of the Association for Computing Machinery (ACM), the oldest, largest, and most prestigious computing society in the world. I am the President of ACM's Special Interest Group on Computer-Human Interaction, one of its largest and most active special interest groups. I also am a member of the ACM Council, the highest governing body of the ACM. I received the ACM for Distinguished Scientist Award in 2009.

15. My research and teaching focus on human-computer interaction, user interface design, and social computing. I have several decades of experience in these specialties of computer science in both industry and academia. I worked for AT&T Laboratories from 1991 through 2002, during which time I conducted research and developed systems that solved problems in software engineering, web information seeking and organization, and recommender systems. In all my research, I designed, implemented, and tested graphical user interfaces. I have been employed full-time as a professor at the University of Minnesota since 2002,

during which time I have taught classes in computer science, human-computer interaction and social computing, and have conducted, supervised, and published research in the field. My research has been published in numerous journal and conference papers, as well as in a book I co-authored entitled “Foundational Issues in Artificial Intelligence and Cognitive Science: Impasse and Solution.” I have served on the editorial board of ACM’s Transactions on Human-Computer Interaction and the Communications of the ACM, and have served as a reviewer for numerous journals, including ACM Computing Surveys, IEEE Transactions on Data and Knowledge Engineering, the International Journal of Human-Computer Studies, and the Journal of Computer-Supported Cooperative Work.

16. I am a listed inventor on nine patents, including those related to the computer graphical environment (U.S. Patent No. 5,680,530).

17. I have consulted on over 10 intellectual property cases that have dealt with topics in user interface design, recommender systems, web information systems, and set top boxes. During these cases, I have testified before judges, been deposed, and written multiple expert reports.

18. A more complete recitation of my qualifications, background and experience, including a list of my publications and matters on which I have worked as an expert, are set forth in my curriculum vitae, attached to my declaration as Appendix A.

B. Compensation

19. I am being compensated by Microsoft for my work in connection with this declaration. The compensation is not contingent upon my performance, the outcome of this post-grant review or any other proceeding, or any issues involved in or related to the post-grant review.

C. Documents and Other Materials Relied Upon

20. My opinions expressed in this declaration are based on documents and materials identified in this declaration, including the '954 patent and its prosecution history, the background materials discussed in this declaration, and any other references specifically identified in this declaration. I have considered these materials in their entirety, even if only portions are discussed here.

21. I have also relied on my own experience and expertise in human-computer interaction and user interface research and design.

22. All exhibit numbers used in this declaration refer to the Exhibits to Petitioners' PGR Petition for the '954 patent.

III. STATEMENT OF LEGAL PRINCIPLES

A. Claim Construction

23. Petitioner's counsel has advised that, when construing claim terms in an unexpired patent, a claim subject to post-grant review receives the "broadest reasonable construction in light of the specification of the patent in which it

appears.” Petitioners’ counsel has further informed me that the broadest reasonable construction is the broadest reasonable interpretation of the claim language, and that any term that lacks a definition in the specification is also given a broad interpretation.

B. Written Description

24. Petitioner’s counsel has advised that, under AIA 35 U.S.C. § 112(a), a patent’s written description must clearly allow persons of ordinary skill in the art to recognize that the inventor invented what is claimed. To satisfy § 112(a), the disclosures in the patent application, as filed, must describe the full scope of the claimed invention and do so in sufficient detail that a POSITA would clearly conclude that the inventor was in possession of the claimed invention as of the filing date.

25. Petitioner’s counsel has advised that it is not enough to show that a claimed invention would have been *obvious* to a POSITA in light of the specification. I have also been informed that, to satisfy § 112(a), it is not enough to point to language in the written description covering separate, individual limitations. For a claim to satisfy § 112(a), the patent’s specification must demonstrate to those a POSITA that the inventor had actual possession of the complete and final invention with all its claimed limitations.

C. Benefit of Claimed Priority / Effective Filing Date

26. Petitioner's counsel has advised that the effective filing date for a patent claim is the actual filing date of the patent, unless the patent claims priority to an earlier application and the claim in question is fully supported under 35 U.S.C. § 112 by the disclosures in the earlier filed application. If the earlier application does not describe the claimed invention in sufficient detail that a POSITA would conclude that the inventor was in "possession" of the claimed invention as of the filing date of that earlier application, the claim does not receive the benefit of its earlier filing date.

D. Anticipation

27. Petitioner's counsel has advised that in order for a patent claim to be valid, the claimed invention must be novel. Petitioners' counsel has further advised that, if the '954 patent is not entitled to claim the benefit of an earlier priority date, the novelty of its claims will be determined by AIA 35 U.S.C. § 102.

28. Petitioner's counsel has advised that, under AIA 35 U.S.C. § 102(a)(1), if a prior art system in public use, on sale, or otherwise available to the public before the effective filing date of the claimed invention satisfies every element of a claim, the claim is anticipated and is not patentable. AIA 35 U.S.C. § 102(b) provides certain exceptions to § 102(a) — for example, where the public use, sale, or public availability of the prior art system was from one of the patent's

named inventors or where the prior art system was developed pursuant to a joint research agreement.

E. Obviousness

29. Petitioner's counsel has also advised me that obviousness under AIA 35 U.S.C. § 103 is a basis for invalidity for patents filed after March 16, 2013, that are not entitled to the benefit of an earlier effective filing date. I understand that if a piece of prior art does not disclose all of the limitations of a given patent claim, that patent claim is nonetheless invalid if the differences between the claimed subject matter and the prior art are such that the claimed subject matter as a whole would have been obvious to a POSITA. Obviousness can be based on a single piece of prior art or a combination that either expressly or inherently disclose all limitations of the claimed invention.

IV. OVERVIEW OF THE '954 PATENT

A. Summary of the '954 Patent

30. The '954 patent was filed on October 27, 2015, and issued on August 23, 2016. It is titled "Graphical User Interface Methods, Systems, and Computer Program Products."

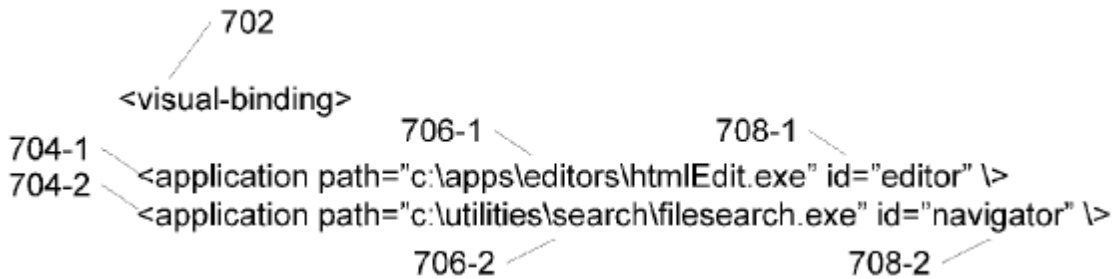
31. In its "Background" section, the '954 patent explains that there is often "no integration and/or cooperation between or among applications used at the same time by a user." *Id.* at 1:23-25. Users have to work in different applications,

and, as a result, “spend significant time managing the user interfaces of these various applications in order to access the data desired in the application desired.” *Id.* at 1:35-37. For that reason, the patent states that “there exists a need for methods, systems, and computer program products for binding attributes between visual components.” *Id.*

32. To address this need, the '954 patent proposes binding visual interface elements within a single application or across different applications through the use of arbitrarily programmable “binding information.” *Id.* at 12:56-13:1. The binding information “specifies a mapping between a first visual attribute of the first visual component and a second visual attribute of a second visual component.” *Id.* at 12:56-62. This “[b]inding information may be represented in any suitable representation including declaratory representations, source code representations, binary representations, and script representations.” *Id.* at 13:24-27.

33. The '954 patent describes in detail how such “binding information” can be defined and “represented in extensible markup language (XML) according to a schema specifying at least one of a format and a vocabulary for defining valid” mappings between application interface elements. *Id.* at 13:21-24. That XML schema provides a “<visual-binding>” element that the programmer can use to define a binding. *Id.* at 13:28-31.

34. Within that “<visual-binding>” element, the programmer can specify the applications involved in the binding using one or more “<application>” tags. *Id.* at 32-35. Figure 7 of the patent depicts a “<visual-binding>” element with “<application>” tags in the following excerpt:



The pair of “<application . . . >” tags within a “<visual-binding>” element identify the two applications whose visual interface elements will be bound, providing their file path by setting a “path” value (*id.* at 32-35) and giving them each an alias by setting an “id” value (*id.* at 40-43). In this example, the binding involves two applications, “an HTML editor application” identified as “editor” and a “file search application” identified as “navigator.” *Id.* at 43-52.

35. The '954 patent describes that mappings between interface elements of different applications are defined in this XML schema for “binding information” through an “<attribute-binding>” tag. *Id.* at 13:63-67 (“An <attribute-binding> tag 710 may be defined for specifying a mapping between a first visual attribute of a first visual component of a first application and a second visual attribute of a second visual component of a second application.”). *Id.* at 13.

36. The “<attribute-binding>” tag takes an “id” value that identifies an visual “attribute” of an interface element of one of the previously defined applications. *Id.* at 13:67-14:3. For example, the patent describes that an “<attribute-binding>” tag with an “id” value of “editor.main.state” would refer to the “visual state” of the “main window” of the previously identified “editor” HTML editor application. *Id.* at 14 :3-16. The ’954 patent also describes “<attribute-binding>” tags with “id” values of “size” (*id.* at 18:49-51, 19:1-3), “location” (*id.* at 19:19-21), and “transparency (*id.* at 19:59-61).” *See also id.*, Fig. 7.

37. The “<attribute-binding>” tag also takes an “op-id” value that is used “to identify a change in an attribute to be detected.” *Id.* at 14:21-24. For example, the ’954 patent describes that an “<attribute-binding>” with an “op-id” value of “init” for the “state” attribute may “indicate a change in a visual component from uninitialized and not visible to initialized and visible.” *Id.* at 14:24-27. An “<attribute-binding>” with an “op-id” value of “change” for the “size” attribute of the main window “may be defined to indicate that any and/or every change in the size of the main window is to be detected.” *Id.* at 18:51-53. The ’954 patent also describes “op-id” values of “max” (“defined to indicate a change of the size of the main window to a maximum size,” *id.* at 18:25-28), “change” (“defined to indicate that any and/or every change in the size of the main window is to be detected,” *id.*

at 18:51-53), “on-focus” (“defined to identify a change including an assignment of input focus for an input device for the visual component,” *id.* at 19:39-42), and others.

38. Finally, within an “<attribute-binding>” tag, the ’954 patent describes that the programmer can specify a “<bind>” tag “defined for specify[ing] a mapping between a visual component identified in an including <attribute-binding> tag 710 by an id attribute 712 and a second visual attribute of a second visual component of a second application.” *Id.* at 14:44-49.

39. In summary, the “<attribute-binding>” tag defines what changes to look for (i.e., identifies a visual attribute of some user interface component to be *monitored* and a specific change in that attribute to be *detected*) and the “<bind>” tag defines what to do when such a change is detected (i.e., identifies a visual attribute of some different user interface component, possibly in another application, and what operation to perform on that attribute). A “<visual-binding>” tag can include numerous “<attribute-binding>” tags, as depicted in Figure 7:



Fig. 7

40. The '954 describes a method and system of *identifying* this programmable binding information and enforcing the mappings that are defined therein. The method is described in Figure 2 and the system in Figure 3:

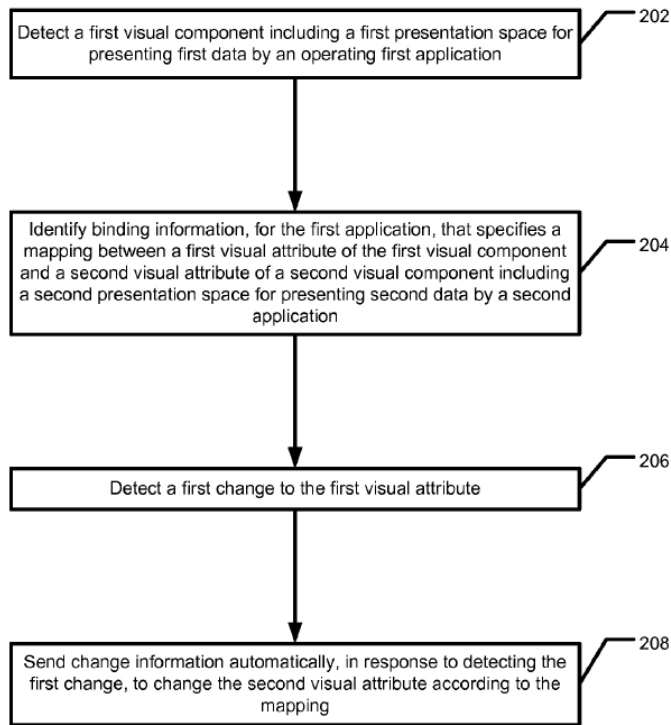


Fig. 2

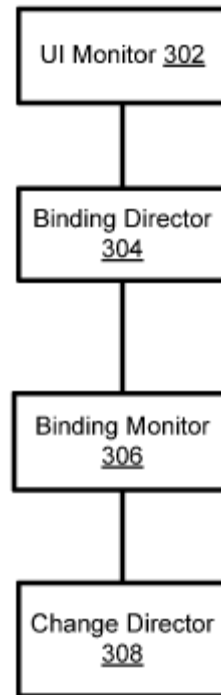


Fig. 3

41. Each component of the system in Figure 3 implements one aspect of the method in Figure 2. Specifically, the “user interface monitor component 302” first detects a “visual component including a first presentation space . . . by an operating first application.” *Id.* at 11:36-39. Binding director 304 then “identif[ies] binding information, for the first application, that specifies a mapping between a first visual attribute of the first visual component and a second visual attribute of a second visual component . . . [of] a second application.” *Id.* at 13:2-7. Binding monitor 206 “detect[s] a first change to the first visual attribute.” *Id.* at 16:10-12. Finally, change director 308, “in response to detecting the first change,

automatically send[s] change information to change the second visual attribute according to the mapping.” *Id.* at 17:36-39.

42. Figures 6a and 6b of the '954 patent describes two simple examples of functionality that a programmer could specify with binding information and implement using the described methods and systems. *Id.* at 20:11-13 (“FIGS. 6a-b illustrate other exemplary mappings that may be supported by various adaptations of the arrangement of components in FIG. 3 in various aspects”).

43. In Figure 6a, “a top border of first app[lication] visual component 604-1a may be mapped to a bottom border of second app visual component 604-2a as specified by binding information received by a binding director component.” *Id.* at 20:13-17; *id.*, Fig. 6:

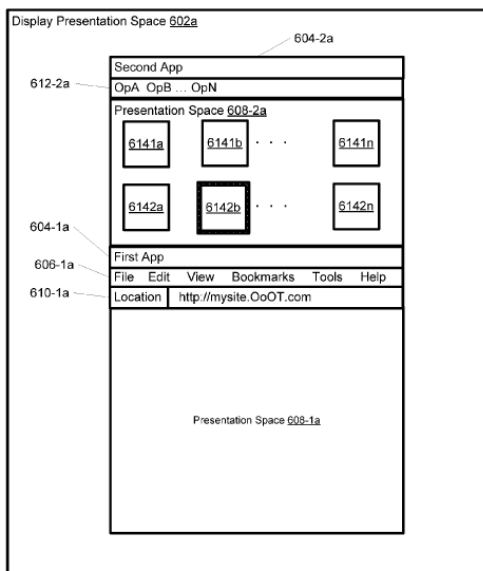


Fig. 6a

In Figure 6a, “binding information may specify that changes to a location and/or size of the top border of first app visual component 604-1a are to be mirrored or matched by changes in the bottom border of second app visual component 604-2a.” *Id.* at 20:46-50. When a “top border” of the first application user interface is raised, the size of the second application user interface “may change. For example, the sider borders may be made shorter.” *Id.* at 20:61-62.

44. In Figure 6b, three separate applications have visual components whose “size and location attributes . . . may be bound.” *Id.* at 21:4-6. When a visual component from the first application “changes from minimized to a restored or maximized size and location, change information may be sent to respective applications by a change director component 408 to change one or both of second app visual component 604-2b and third app visual component 604-3b to a minimized state presented in specified respective locations in display presentation space 602b.” *Id.* at 21:3-13; *id.*, Fig. 6b:

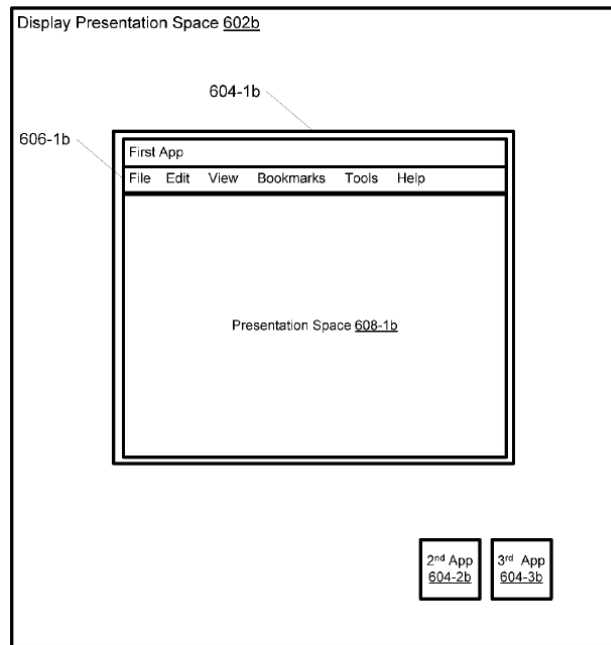


Fig. 6b

45. The focus of the '954 patent is on *how* to specify binding information using the disclosed XML schema and on the various components of the system it describes to *identify* the specified binding information and corresponding user interface components (using the “UI Monitor” and the “Binding Director”) and then *enforce* the mappings in the identified binding information (using the “Binding Monitor” and “Change Director”). The written description makes clear that the invention is *not* directed or limited to specific user interface functionality. *See, e.g.*, 5:65-6:8 (describing a “not exhaustive” list of over twenty different “visual interface elements,” including “windows, textboxes, sliders, list boxes, drop-down lists, spinners, various types of menus, toolbars, ribbons, combo boxes,

tree views, grid views, navigation tabs, scrollbars, labels, tooltips, text in various fonts, balloons, dialog boxes, and various types of button controls including check boxes and radio buttons”); 16:31-35 (describing thirteen “exemplary” visual attributes “that may be included in a mapping specified in binding information, includ[ing] one or more of a font, a color, a location in a presentation space, a size, a visibility attribute, a shape, a measure of time, a Z-value, a transparency attribute, a shading attribute, a visible pattern, a line thickness, and an attribute of a background.”). A person of skill in the art would understand the invention described in the ’954 patent to be a system that allows programmers to specify “binding information” that relates visual attributes of different applications to each and that identifies that binding information and enforces the mapped relationships specified therein.

46. The only user interface functionality disclosed in the ’954 patent is in the context of the programmable binding information. The only changes to a user interface that are described in the ’954 patent are the changes that can be specified by the programmer in the specified binding information for *detection* by the described system or that can be specified in the binding information as a *result* of some other detected change. The ’954 patent does not purport to describe any novel user interaction outside of that context.

47. The '954 patent does not disclose any ordered sequence of steps in which consecutive user inputs cause consecutive modifications to the user interface. Indeed, the '954 patent specifically seeks to remove the need for sequences of user interaction, which cause the “[u]sers [to] spend significant time managing the user interfaces of these various applications.” *Id.* at 1:35-37.

B. Prosecution History

48. I understand that the application that issued as the '954 patent on October 27, 2015. I understand the '954 patent is a continuation of App. No. 14/331/096 (“the '096 application,” still pending), Ex. 1015, which is a continuation of U.S. Pat. No. 8,780,130 (Ex. 1006, “the '130 patent”), filed as App. No. 12/956,008, Ex. 1016.

49. With the exception of the summary sections—each of which parrots the claims filed with its respective application—the disclosures of each of the '954 patent application, the '096 application, and the '130 patent are substantively identical.

50. I understand that the claims of the '954 patent were not filed with the original application. On January 31, 2016, the applicant cancelled each of the original claims in the '954 application and added an entirely new set of claims. Ex. 1007 at 12-23. Additional claims were added a few months later. *Id.* at 24-35.

51. None of these later-claims to the '954 application relate to “binding information” or the method or system disclosed in the patent’s written description and figures. '954 patent, claims 1-20.

C. Related District Court Litigation

52. I understand that Cypress Lake has filed nine lawsuits alleging infringement of the '954 patent against various manufacturers of devices that run Windows 10. I understand that Cypress Lake has submitted infringement contentions asserting infringement of claims 1-23 and 25-31 and provided claim charts alleging that the capabilities provided by Microsoft’s Windows 10 operating system, and in particular the “Snap Assist” functionality, satisfy each and every limitation of the asserted claims. I have reviewed these claim charts.

V. STATE OF THE ART PRIOR TO THE '954 PATENT

53. Graphical User Interfaces (“GUIs”) have been the primary model of computer-user interaction since the 1980s . There is long, rich history over the past four decades of research and commercial systems exploring graphical user interfaces (“GUIs”).

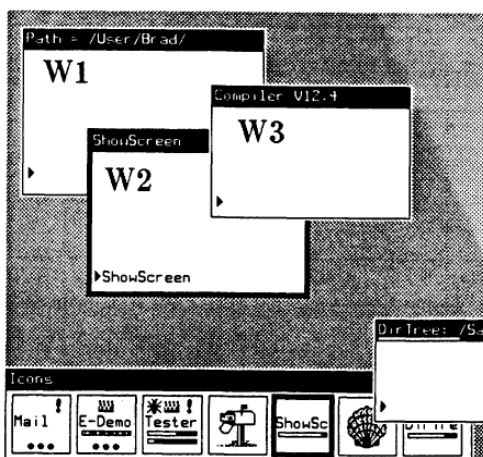
A. Correlated User Interface Elements

54. The '954 patent is directed to using programmable “binding information” to create mappings between visual components of different applications. Outside the context of programmable “binding information,” the

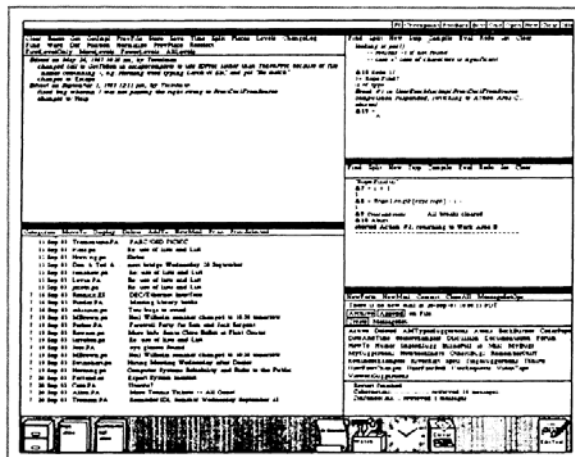
basic concept having a user interface element in one application respond to a change in a user interface element of another application was well known and commonly used in operating systems for over 30 years. Indeed, both of the specific examples of functionality described in the figures of the '954 patent were used in the very first version Microsoft Windows.

55. Most GUIs, especially on larger screens, include a notion of “windows,” which group and display information into (usually rectangular) portions of the screen that can be opened, closed, moved, resized, rearranged, minimized, etc. Windows can be associated with specific applications or with the operating system itself. An application running on a computer with a GUI might have no associated windows, a single associated window, or several associated windows. For example, many applications use a “document” model, where each open document in an application is present in its own separate window.

56. Different operating systems have used a variety of different windowing models. For example, many GUI systems employed a notion of “overlapping” windows, creating the appearance that some windows were above other windows, whereas others used a notion of “tiling” windows where Windows that could be rearranged and resized together but could not overlap. Ex. 1032, pp. 67 & 71-73.



“Sapphire” window manager with overlapping windows, Ex. 1032, Fig. 8.



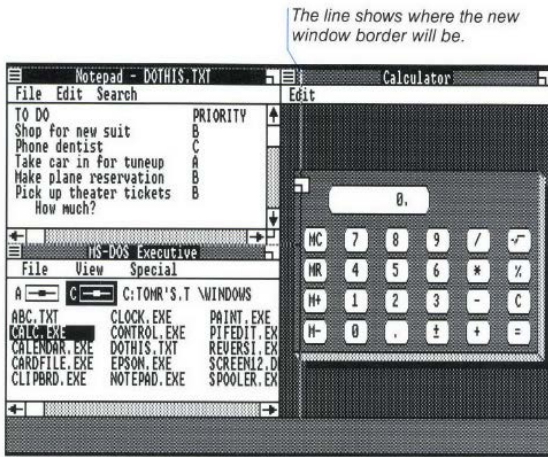
“Cedar” window manager with tiled windows, Ex. 1032, Fig. 2.

Allowing windows to overlap can provide more flexibility to the user, but it can also create clutter and require more explicit organization from the user (e.g., managing how the windows are ordered front-to-back). *Id.* at 67 & 71-73. In contrast, tiling windows can maximize screen space and reduce complexity and clutter, but can prove difficult on small screens and provide less freedom overall. *Id.* at 67 & 71-73. Many window systems over the years have adopted a hybrid approach — freely allowing windows to overlap but providing tools to organize windows in a tiled manner — thereby providing benefits from both approaches.

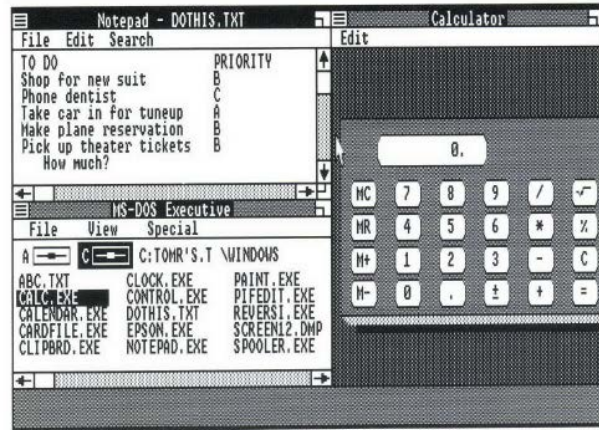
57. Windows 1.0 was released in 1985 and used a tiling windowing model. These tiled windows shared a border, and moving that border to resize one window would result in other displayed windows being resized accordingly, exactly like the functionality described in the '954 patent's Figure 6a. 1020 at 49-50:

DECLARATION OF DR. LOREN G. TERVEEN
IN SUPPORT OF PETITION FOR POST-GRANT REVIEW
OF U.S. PATENT NO. 9,423,954

5. Drag the size pointer to the right until the window is the size you want and release the mouse button.

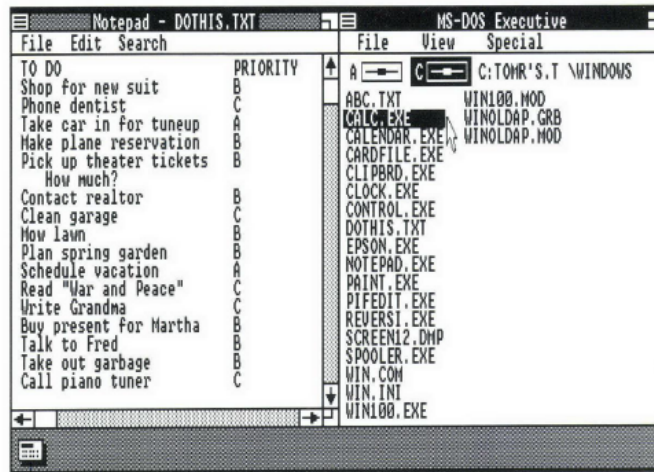


Windows widens the Notepad window and adjusts the size of any adjacent windows.



This is the exact functionality that the '954 patent describes for Figure 6a. '954 patent at 20:47-50 (“changes to a location and/or size of the top border of first app visual component 604-1a are to be mirrored or matched by changes in the bottom border of second app visual component 604-2a”).

58. Windows 1.0 also provided functionality to “minimize” open windows, displaying them as icons in the “icon area” instead of as windows in the “work area.” Ex. 1020 at 44:



The Calculator icon appears in the icon area.

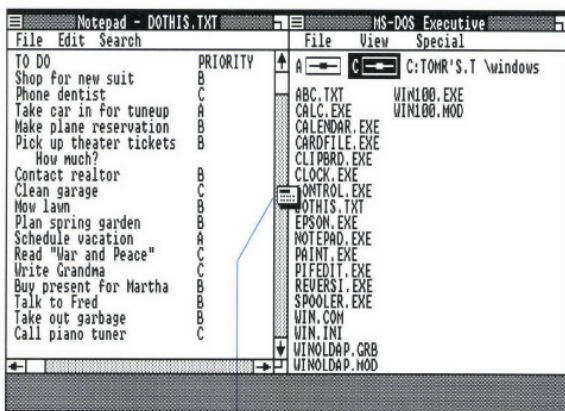
The Calculator icon appears in the icon area. The Calculator is running in memory, but isn't taking up space in the work area. Now you can expand the Calculator icon and position your windows in a variety of ways on the screen.

Windows 1.0 allowed the user to drag icons from the “icon area” up to the work space to maximize them into full-sized windows. *See Ex. 1020 at 45-46:*

- 1 Point to the Calculator icon in the icon area.
- 2 Press the mouse button.
- 3 Drag the Calculator icon onto the border between the Notepad and MS-DOS Executive windows.

- 4 Release the mouse button.

Note that the mouse pointer changed shape.

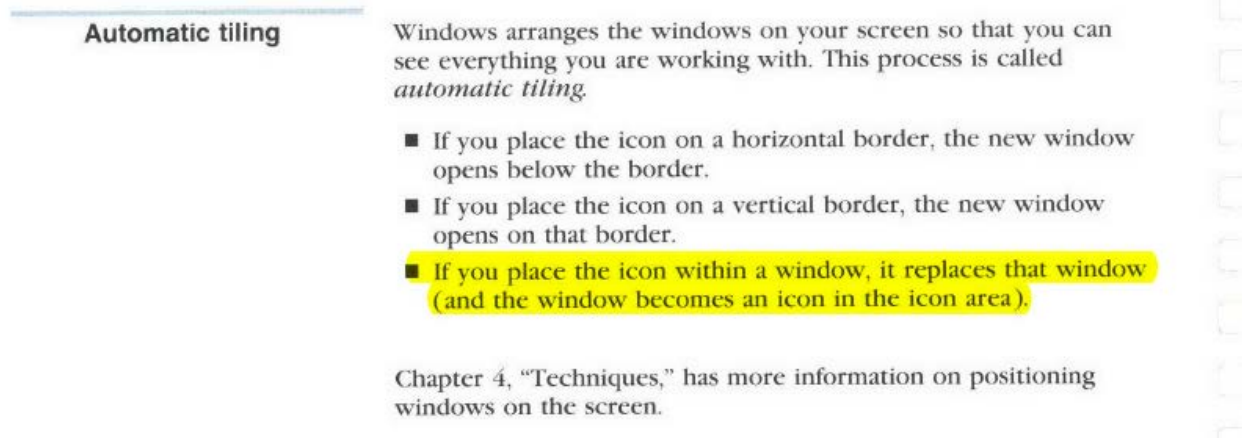


Position the Calculator icon on the border.



The Calculator window expands in the work area. The windows to the left and right are adjusted.

If the user placed the icon on top of an existing window, rather than on the border, the previously iconized window would be maximized and replace the existing window, which would be minimized. *Id.* at 46 (highlighting added):



Thus, Windows 1.0 has the exact user interface functionality that the '954 patent described for Figure 6b. *See* '954 patent at 21:6-11 (“When first app visual component 604-1 b changes from minimized to a restored or maximized size and location, change information may be sent to respective applications . . . to change one or both of second app visual component 604-2 b and third app visual component 604-3 b to a minimized state . . .”).

59. The idea of changing an interface element in one application in response to a change in a different application’s interface was well known in the art in 2010. In fact, the exact functionality described in the '954 patent was in common use in Microsoft’s Windows 1.0 operating system 25 years before Cypress Lake’s first priority application was filed. The only arguable difference

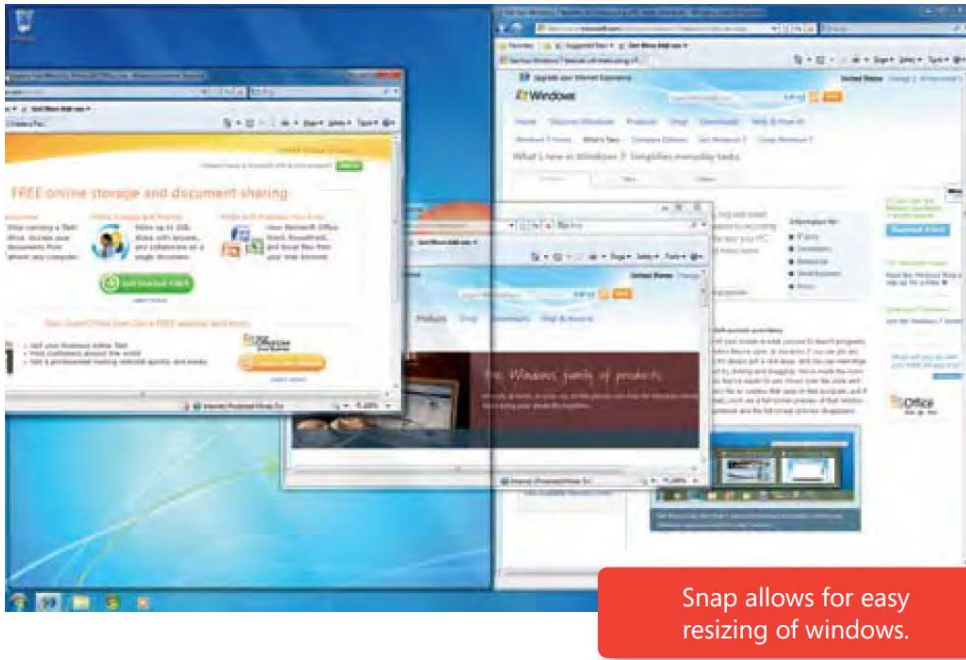
between what is described in the '954 patent and basic functionality that was common in the art is the specific mechanism of achieving that functionality through the use of programmable binding information.

B. Window Layout via “Snapping”

60. I understand Cypress Lake has accused Microsoft of infringing the '954 patent based on the “Snap Assist” feature in Windows 10, which is for laying out windows on screen, not for navigating *between* them. Windows 10 with Snap Assist was released in July 2015 and was therefore in public use prior to the filing of the '954 patent. *See* Ex. 1008 (“Sundelin Dec.”) ¶¶ 19-25.

61. The Snap Assist feature in Windows 10 builds on several similar features in earlier Windows operating systems. Windows 7 introduce the “Aero Snap” feature in 2009 that allowed users to easily lay out windows in useful layouts. Dragging a window to the top of the screen would make it fill the entire display. Dragging a window to the side of the screen would “snap” it to that side, making it easy to layout windows side-by-side by dragging simply dragging each window to the desired side of the screen. *See* Ex. 1021 at 33:

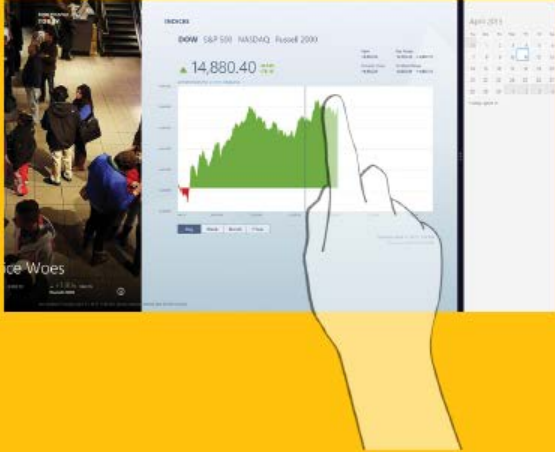
Do you sometimes compare the contents of two different windows? That's easy too. When you drag the top of each window to opposite sides of the screen, as your cursor touches the edge, the window will resize to fill that half of the screen.



62. Windows 8 included a similar feature, but with a different windowing model. Like Windows 1.0 before it, Windows 8 primarily used a “tiling” model where windows could be displayed side-by-side but could not overlap. *See* Sundelin Dec. ¶ 9. As in Windows 7, windows could be “snapped” side-by-side, but in a tiled context. *See* Ex. 1022 at 15:

Snap an app: Use two apps at once.

Keep an eye on a certain company's stock you're following while you work on your proposal for them. You can snap a second app to the left or right of your screen while you're using a different one, making it easy for you to get more done.*



With touch, slide your finger from the left edge to bring in the second app. Drag that app to the left or right side of the screen until an opening appears behind it.

With a mouse, move your pointer to the upper-left corner until the second app appears, and then drag that app to the left or right of the screen until an opening appears behind it.

Managing Apps

* Your screen resolution must be at least 1366 x 768 to snap apps.

As in other tiling window systems, the user could resize side-by-side windows in Windows 8 by dragging the joint divider that separated them. *See Sundelin Dec. ¶¶ 10.*

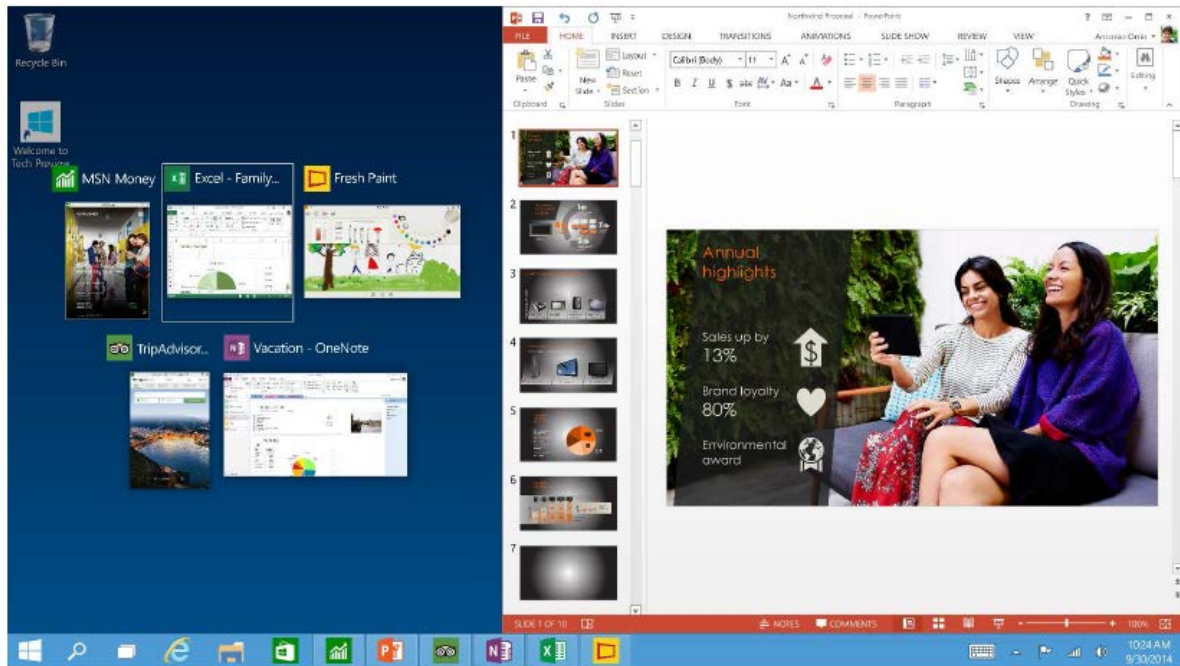
63. Announced in 2014 and released in July 2015, Windows 10 introduced two different windowing modes — “Tablet Mode,” which uses a tiling window approach like Windows 8, and “Desktop Mode,” which uses overlapping windows like in Windows 7. *See Sundelin Dec. ¶¶ 24.* Both Tablet Mode and Desktop Mode included a new “Snap Assist” feature that built on Windows 7’s Aero Snap. *Id.*, ¶¶ 20. As in 2009 with the Windows 7 “Aero Snap” feature, the user could “snap” a window to either side of the screen. *Id.*, ¶¶ 20. However, once

the user “snapped” a window, Windows 10’s Snap Assist would now display thumbnails of other application windows to let the user more quickly select another window to fill the remaining space. Ex. 1028 at 6:

Multitasking

3.1 Snap windows

Working in multiple apps and programs is easier and more intuitive thanks to enhancements in snapping. Have up to four programs snapped on the same screen; Windows will even suggest how to fill the gaps with other open apps and programs.



To snap, click and hold the top of the window and drag it to either the side or corner of the screen. You’ll see a shadow of where the window will snap; release the window to snap it into place. Note that in Tech Preview, you are able to view up to four programs at the same time; Windows Store apps can only be snapped into half screen.

C. The Level of Ordinary Skill in the Art

64. I understand from Petitioners’ counsel that the claims and specification must be read and construed through the eyes of a POSITA at the time of the priority date of the claims. I have also been advised that to determine the

appropriate level of ordinary skill in the art, the following factors may be considered: (a) the types of problems encountered by those working in the field and prior art solutions thereto; (b) the sophistication of the technology in question and the rapidity with which innovations occur in the field; (c) the educational level of active workers in the field; and (d) the educational level of the inventor.

65. The '954 patent relates to using programmable "binding information" to create mappings between visual interface elements in different applications.

66. Based on the above considerations and factors, it is my opinion that a person of ordinary skill in the art at the time of the purported invention would have had at least a bachelor's degree in electrical engineering, computer science, or interaction design and three to five years of experience in user interface design. A recipient of other science or engineering degrees may qualify as a POSITA if they have taken relevant coursework or have experience to understand and contextualize the user interface subject matter at issue. This description is approximate and additional educational experience could make up for less work experience and vice versa. As evidenced by my experience and education discussed in Sec. II(A), I qualify as a person of at least ordinary skill in the area of the '954 patent.

67. The level of ordinary skill in the art would not have meaningfully differed between 2010 (the earliest date to which Cypress Lake claims priority) and 2016 (when the claims of the '954 patent were added to the application).

VI. IDENTIFICATION OF THE PRIOR ART AND SUMMARY OF OPINIONS

68. It is my opinion that claims 1-20 of the '954 patent are unpatentable for lack of written description support because they retroactively claim inventions that a person of ordinary skill in the art would not understand to have been disclosed in the specification. A POSITA would not understand from the '954 specification or the incorporated applications that the named inventors were in possession of the complete and final invention covered by the claims.

69. It is also my opinion that if the effective filing date of the claims of the '954 patent is the actual filing date of October 27, 2015, claims 1-2, 4-6, 8, 10-15, and 16-20 of the '954 patent are rendered obvious by prior use and sale of Microsoft Windows 10 under the interpretation of the claims that Cypress Lake has advanced in parallel district court proceedings.

70. It is my opinion that claims 1-20 of the '954 patent are invalid based on the following grounds:

Ground 1	Claims 1-20 lack written description support under 35 U.S.C. § 112(a)
-----------------	---

Ground 2	Claims 1-2, 4-6, 8, 10-15, and 16-20, as interpreted by Cypress Lake in district court proceedings, are rendered obvious under 103 by the public use and sale of Microsoft Windows 10 and devices running Windows 10.
-----------------	---

VII. UNPATENTABILITY OF THE CHALLENGED CLAIMS OF THE '954 PATENT

A. Ground 1: Claims 1-20 Are Invalid Under 35 U.S.C. § 112(a) For Lack Of Written Description

1. Independent Claims 1, 14, and 19

71. The '954 patent's three independent claims 1, 14, and 19, are each directed to an apparatus with a processor configured for performing a specific, ordered sequence of steps, each step consisting of detecting a "user input" and modifying the user interface responsive to that "user input" in some particular manner.

72. Claim 1 is representative and recites:

1. An apparatus, comprising:

at least one processor configured for coupling with a display, memory, and at least one input device and further configured for:

storage of a first application, a second application, and a third application, utilizing the memory;

detection of a first user input, utilizing the at least one input device;

in response to the first user input, presentation of, utilizing the display, a plurality of reduced windows that do not visually overlap each other, the plurality of reduced windows including

a second application reduced window associated with the second application and a third application reduced window associated with the third application which are both presented exterior to a first window associated with the first application without visual overlap between the first window and the plurality of reduced windows;

detection of a second user input in connection with the second application reduced window associated with the second application, utilizing the at least one input device;

in response to the second user input in connection with the second application reduced window associated with the second application, **presentation of**, utilizing the display, **a second window associated with the second application** for presenting second data associated with the second application, **adjacent to the first window** associated with the first application;

detection of a third user input, utilizing the at least one input device;

and in response to the third user input, change of, utilizing the display, the presentation of the first window and the second window, such that a first size of the first window and a second size of the second window are both changed.

(Emphasis added.)

73. Claim 14 is similar to claim 1 with respect to the highlighted limitation. See '954 patent at 27:52-60. Claim 14 further requires that the changed-size application windows remain adjacent and non-overlapping after the third user input. See '954 patent at 27:66-67.

74. Claim 19 adds fourth, fifth, and sixth steps to the sequence of detected “user inputs” and resulting modifications. *Id.* at 29:10-30:15.

a. No Written Description Support for Presenting a Second Window Adjacent to a First Window In Response To User Input

75. Claims 1, 14, and 19 all recite detecting a user input, and, in response to a user input, presenting a new application window “adjacent” to an application window that is already displayed. *See, e.g.*, ’954 patent, claim 1 (“in response to the second user input ... presentation of ... a second window associated with the second application for presenting second data associated with the second application, adjacent to the first window associated with the first application”).

76. A POSITA would understand the specification to describe an invention for specifying programmable “binding information” and a system for identifying that binding information and enforcing the mappings its defines by monitoring user interface components for specific changes and sending change behavior to the other user interface components identified in the mappings when the specified changes are detected. The written description would not convey to a POSITA that the inventor was in possession of any invention comprising a sequence of user inputs where a second (or third) user input results in “presenting” a new application window “adjacent to” a first application window. Nothing in the written description of the ’954 patent would suggest to a POSITA that the inventor was in possession of an invention comprising functionality not specifically implemented via programmable binding information.

77. The written description describes displaying application windows adjacent to one another in only two instances. In both, as might be expected, the windows' positioning is entirely dependent on previously identified programmable binding information explicitly requiring the visual components to be displayed adjacent to one another.

78. Figure 6a and the accompanying text disclose an “exemplary mapping” where “a top border of first app visual component 604-1a may be mapped to a bottom border of second app visual component 604-2a as specified by binding information” *Id.* at 20:15-17 (emphasis added); *id.*, Figure 6a:

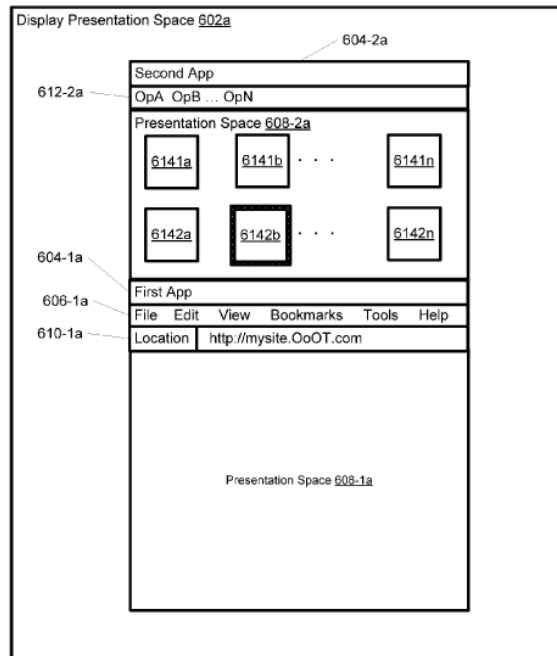
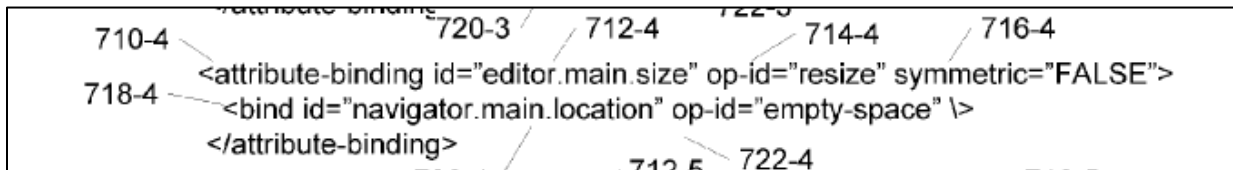


Fig. 6a

This functionality described was specifically in the context of the programmable “binding information” described in the rest of the ’954 patent. *See, e.g.*, Fig. 7. The text accompanying Fig. 6a explains that “*binding information* may specify that changes to a location and/or size of the top border of first app visual component 604-1a are to be mirrored or matched by changes in the bottom border of second app visual component 604-2a.” *Id.* at 20:46-50. The mirroring functionality can be defined in the programmable “binding information” either by “mapping the two borders” or by “mapping size and location attributes between the two visual components.” *Id.* at 20:50-53; *see also id.* at 20:58-62. If a newly-presented application window is caused to be positioned adjacent to another window in this description, it must be because of the mappings in the identified binding information, not because of some arbitrary user input.

79. Figure 7 and the accompanying text similarly describe using binding information to define a mapping between the sizes of visual components. *Id.* at 19:3-5 (“Op-id attribute 714-4 specifies the value ‘resize’, which may be defined to be an equivalent of the ‘change’ value described above.”); *id.*, Figure 7:



```
<attribute-binding id="editor.main.size" op-id="resize" symmetric="FALSE">
  <bind id="navigator.main.location" op-id="empty-space" \>
</attribute-binding>
```

Annotations in the image: 710-4 points to the opening tag, 718-4 points to the opening tag of the inner bind element, 714-4 points to the op-id="resize" attribute, and 716-4 points to the symmetric="FALSE" attribute. Other annotations include 720-3, 712-4, 722-4, and 712-5.

Using this “op-id attribute,” which is specified as “empty-space” in Figure 7’s example, a programmer may define binding information for a component “to indicate that the main window of the navigator application is to be changed to fill the largest space external to the first visual component identified in <attribute-binding> tag 710-4 in a presentation space including both visual components.” *Id.* 19:11-16. Again, this will only result in a second application window being presented adjacent to a first one, if a system described in Fig. 3 had previously identified explicitly defined binding information, mapping the visual components with the appropriate “op-id attribute” information. The binding information, not some arbitrary user input, is what would cause a new application window to be presented adjacent to an existing one.

80. The “binding information” itself would not be understood to be a “user input” — it is representation of information (defining mappings between arbitrary visual components), not an input. *Id.* at 13:21-22 (“binding information [is] represented in extensible markup language (XML)”); *id.* at 13:24-28 (“Binding information may be represented in any suitable representation including declaratory representations, source code representations, binary representations, and script representations, to name a few examples.”). The “binding information” is information that defines relationships between different aspects and components of a user interface relating to *how* those components will behave when used by a

user. A POSITA would understand that is a fundamentally different concept than the inputs that the “user” provides via input devices when actually *using* that interface in the midst of operating a device. The claim term “user input” would be understood by its plain meaning, which a POSITA would understand as covering input provided by a user through an input device. E.g., ’954 patent at 4:64-5:2 (“Execution environment 102 may receive user-provided information via one or more input devices . . . [and] may include an input device adapter for a keyboard, a touch screen, a microphone, a joystick, a television receiver, a video camera, a still camera, a document scanner, a fax, a phone, a modem, a network interface adapter, and/or a pointing device, to name a few exemplary input devices.”).

81. The same is true for each of the other limitations in the ’954 patent claims that recite modifying application windows in response to “user input.” E.g., ’954 patent at 29:17-20 (“in response to the fifth user input, presentation of a third window so as to overlie only some of the first window and only some of the second window”), 25:10-12 (“in response to the first user input, presentation of . . . a plurality of reduced windows”), 25:33-37 (“in response to the third user input, change of . . . the presentation of the first window and the second window such that” sizes are changed), 27:52-55 (“in response to the second user input . . . creation of a second window associated with the second program component”). There is no support in the specification for any of these user interface changes

being performed in response to anything other than previously identified binding, much less some specific “user input” in an arbitrary sequence of user interface actions.

82. Even if “user input” were interpreted broadly enough to encompass the binding information itself, the specification fails to disclose these particular ordered sequences of user inputs and responses. The ’954 patent notes generally that a user input might be involved in the *identification* of binding information that the system will use to enforce the mappings it specifies, *e.g.*, *id.* at 23:1-4 (“Binding information may be identified via a message received via a network, an interprocess communication mechanism, a shared region of a data storage medium, and/or a user input.”); *id.* at 17:7-8, but there is no disclosure of an invention based on user inputs outside the context of binding information, much less of repeated detection of a specific *sequence* of user inputs being used to sequentially modify specific visual components. And even if “user input” did include binding information, the latter would be far narrower than the former. A POSITA would not understand the ’954 patent’s descriptions of using programmable binding information to define mappings between user interface components to suggest any broader invention relating to manipulating user interfaces directly through the use of “user input.” And a POSITA would certainly not understand the ’954 patent’s

description to suggest any specific ordered sequences of user interface manipulations.

83. The patent consistently and exclusively describes the presentation of adjacent application windows as being accomplished through programmable binding information that explicitly specifies mappings of arbitrary visual components across applications. That is what a POSITA would have understood to be the central feature of the invention. The independent claims of the '954 patent ignore this critical feature in all of the patent's disclosures, instead identifying arbitrary sequences of "user inputs" with no support in the written description.

84. For example, claim 1 recites a three-part sequence: in response to a *first user input*, the apparatus presents a first application window and multiple non-overlapping reduced-size windows; in response to a *second user input*, the apparatus presents a second application window next to the first one; and *in response to a third user input*, the apparatus changes the size of these two application windows. '954 patent at 25:2-37. Claim 14 recites a similar sequence. *Id.* at 27:37-67. Claim 19 also recites a similar sequence but adds fourth, fifth, and sixth modifications to application windows in response to sequential user inputs. *Id.* at 29:10-30:15.

85. The only functionality described in the '954 specification is in the context, and as a result of, binding information that has been explicitly specified

and identified by the described system. The '954 patent makes clear that this binding information system can be used map a change in *any* attribute of *any* interface element to *any* other attribute of *any* other interface element. *See, e.g.*, 5:65-6:8 (describing a “not exhaustive” list of over twenty different “visual interface elements,” including “windows, textboxes, sliders, list boxes, drop-down lists, spinners, various types of menus, toolbars, ribbons, combo boxes, tree views, grid views, navigation tabs, scrollbars, labels, tooltips, text in various fonts, balloons, dialog boxes, and various types of button controls including check boxes and radio buttons”); 16:31-35 (describing thirteen “exemplary” visual attributes “that may be included in a mapping specified in binding information, includ[ing] one or more of a font, a color, a location in a presentation space, a size, a visibility attribute, a shape, a measure of time, a Z-value, a transparency attribute, a shading attribute, a visible pattern, a line thickness, and an attribute of a background.”). The '954 patent also makes clear that in “detecting” changes to user interface elements to enforce the identified mappings, it is agnostic as to what causes those changes and how they are “detected.” *See, e.g.*, '954 patent at 16:17-27 (changes to components can be detected by “receiving a request to change the visual component including the attribute, receiving a request for a resource for changing the visual attribute”); *id.* at 23:19 (“receiving a change indication identifying the change”); *id.* at 23:20-23 (“receiving change information via a message received

via a network, an interprocess communication mechanism, a shared region of a data storage medium, and/or a user input”); *id.* at 23:24-28 (“receiving an indication to change a visual attribute, detecting an access to a resource for changing a visual attribute, intercepting a communication for changing a visual attribute, . . . receiving a message via a network identifying a visual attribute”); *id.* at 23:37-39 (“receiving a user input corresponding to the visual component”). The specification provides no implementation details for any purported detection technique and indicates nothing about how to *use* any information “received” to change a visual component. A POSITA would not recognize any of these statements as expressing a particular feature of some other invention in the inventor’s possession. Rather, a POSITA would recognize these statements merely as expressing possible features of or ways of using the specifically disclosed systems for specifying, identifying, and enforcing “binding information” between visual components.

86. These vague, broad statements strewn throughout the specification cannot be the basis for allowing Cypress Lake to retroactively claim arbitrary sequences of steps for *detecting* some specific “user input” and, in response to that user input, presenting or modifying some arbitrary user interface components. No such sequences are disclosed in the specification — particularly when the claims are directed to pure functionality and completely divorced from the “binding

information” at the core of the ’954 patent. Otherwise, Cypress Lake could retroactively claim literally *any* sequence of user interactions and corresponding system responses. A POSITA would not understand the ’954 specification to convey “possession” of any such invention.

87. The only invention that the ’954 patent would clearly convey to a POSITA was in the inventor’s possession is a system that identifies programmed binding information that specifies particular mappings between application user interface elements, and that enforces such mappings by monitoring visual interface elements for the specified changes and, when one is detected, implementing the corresponding change in the other specified interface element. *See, e.g.*, ’954 patent, Figs. 2, 3, 7. Nothing in the ’954 patent would convey to a POSITA that the inventor was in possession of an invention for sequentially presenting and/or modifying the appearance of application windows in response to a series of user inputs, much less to the specific sequences of user inputs and system responses recited by claims 1, 14, and 19.

2. Dependent claims 2-13, 15-18, and 20

88. Claims 2-13 each depend from independent claim 1. Claims 15-18 depend from claim 14. Claim 20 depends from independent claim 19.

89. These dependent claims incorporate all of the limitations of the independent claims from which they depend. None of the dependent claims

narrow the scope of the independent claim limitations in a way that would correct any of the deficiencies identified in the previous section.

90. The '954 written description would not clearly convey to a POSITA that the inventor was in possession of any of those claimed inventions. Thus, each of the dependent claims lack written description support and are invalid under § 112(a) for the reasons discussed with respect to the independent claims in Section VII(A)(1).

B. Ground 2: Claims 1-2, 4-6, 8, 10-15, and 16-20 Are Invalid Under 35 U.S.C. § 102(a)(1) as Anticipated By The Public Use and Sale of Microsoft's Windows 10 Operating System and Devices Running Windows 10

91. I understand that Cypress Lake has filed an action in the Eastern District of Texas against manufacturers of Windows 10 devices. In that action, it has alleged devices running Windows 10 infringe claims 1-2, 4-6, 8, 10-15, and 16-20 of the '954 patent based on Snap Assist and other user interface features.

92. For the reasons described above in Section VIII(A), it is my opinion that neither the '954 specification nor the applications to which it claims priority would clearly convey to a POSITA that the inventor was in possession of the invention claimed. Therefore, I understand that none of the claims of the '954 patent are entitled to an earlier effective filing date.

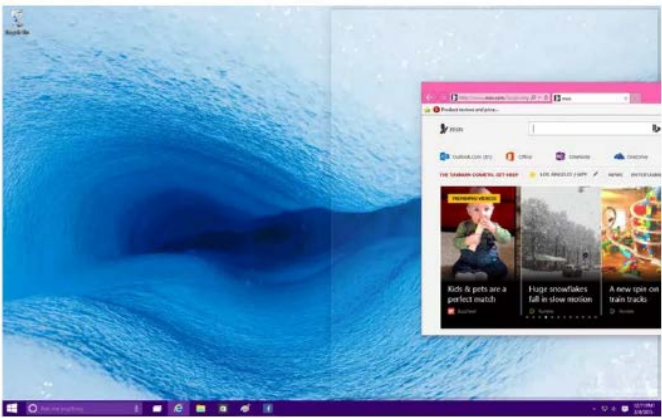
93. I have reviewed the claim charts that Cypress Lake has provided in

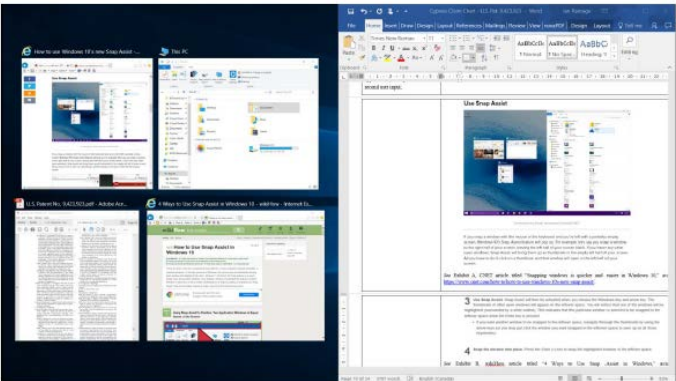
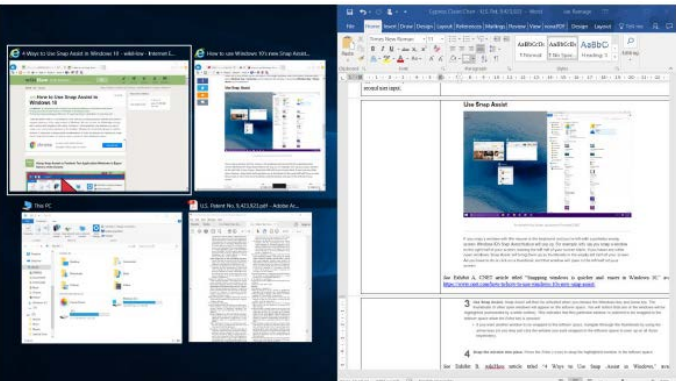
that litigation, which assert that devices running Windows 10 '954 patent claims 1-2, 4-6, 8, 10-15. I understand that devices running Windows with this same or substantially similar functionality were in public use, on sale, and otherwise available to the public before the effective filing date of the '954 patent. Sundelin Dec. ¶¶ 19-25. So the use that Cypress Lake alleges to infringe pre-dates the effective filing date of the '954 patent. I understand that any product that would infringe a claim later in time anticipates that claim if earlier in time. Therefore, it is my opinion that the functionality that Cypress Lake has accused of infringement must in fact anticipate or render obvious those same asserted claims under the very interpretation of the claims that Cypress Lake has promoted in district court.

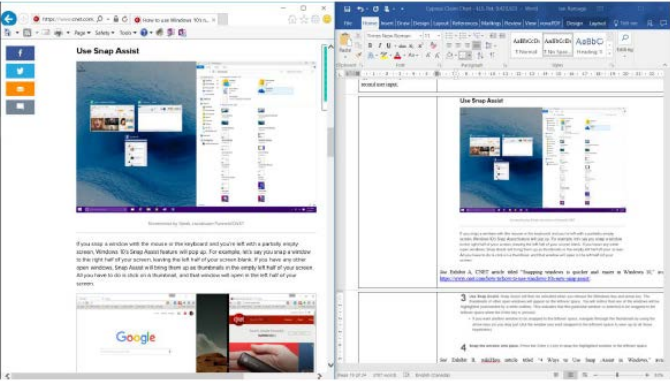
1. Independent Claim 1


94. Cypress Lake alleges that systems running Windows 10 meet each of the limitations of claim 1 of the '954 patent. By Cypress Lake's own interpretation of this claim, each limitation would have been rendered obvious by the Windows 10 devices already in public use, on sale, and otherwise available to the public prior to the effective filing date.

95. Cypress Lake's claim charts assert the following infringement theories:

Claim Limitation	Cypress Lake allegations re. Windows 10
<p>1(a)</p> <p><i>An apparatus, comprising:</i></p> <p><i>at least one processor configured for coupling with a display, memory, and at least one input device</i></p>	<p>Cypress Lake accuses devices “running the Windows 10 Operating System” with “at least one processor” “configured to connect to a display,” “memory (RAM and hard drive,” and “at least one input device (mouse, keyboard, touchpad and/or touchscreen).” Ex 1009 at 1.</p>
<p>1(b)</p> <p><i>and further configured for: storage of a first application, a second application, and a third application, utilizing the memory;</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation because they “can store three (or more) applications in . . . memory.”) <i>Id.</i> at 1.</p>
<p>1(c)</p> <p><i>detection of a first user input, utilizing the at least one input device;</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation because they “can detect a user input . . . to ‘snap’ an application window to either side of the screen” by dragging a window by its title bar to either side of the screen. <i>Id.</i> at 1-3.</p> <div style="text-align: center;">  </div> <p><i>Id.</i> at 2.</p>
<p>1(d)</p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation by “activat[ing]</p>

Claim Limitation	Cypress Lake allegations re. Windows 10
<p><i>in response to the first user input, presentation of, utilizing the display, a plurality of reduced windows that do not visually overlap each other, the plurality of reduced windows including a second application reduced window associated with the second application and a third application reduced window associated with the third application which are both presented exterior to a first window associated with the first application without visual overlap between the first window and the plurality of reduced windows;</i></p>	<p>the ‘Snap Assist’ feature and display[ing] a menu of thumbnails . . . in an area of the screen opposite to where the user snapped the window” <i>Id.</i> at at 3-5.</p>  <p><i>Id.</i> at 5.</p>
<p>1(e) <i>detection of a second user input in connection with the second application reduced window associated with the second application, utilizing the at least one input device;</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation by the user “then choos[ing] (with mouse, touchpad, keyboard, or touch screen) . . . the thumbnail of the second application . . . that he wants to display beside the first one.” <i>Id.</i> at at 5-7.</p>  <p><i>Id.</i> at 7.</p>

Claim Limitation	Cypress Lake allegations re. Windows 10
<p>1(f)</p> <p><i>in response to the second user input in connection with the second application reduced window associated with the second application, presentation of, utilizing the display, a second window associated with the second application for presenting second data associated with the second application, adjacent to the first window associated with the first application;</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation by “display[ing] the second application window in the space next to the first window .” <i>Id.</i> at 7-9.</p>  <p><i>Id.</i> at 9.</p>
<p>1(g)</p> <p><i>detection of a third user input, utilizing the at least one input device; and</i></p> <p><i>in response to the third user input, change of, utilizing the display, the presentation of the first window and the second window, such that a first size of the first window and a second size of the second window are both changed.</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy these limitations because the “user may then select the vertical border between the two windows and drag it left or right to re-size the second window relative to the first . . . ,” and the Windows 10 device “will then resize the windows on the screen accordingly.” <i>Id.</i> at 9-12.</p>

Claim Limitation	Cypress Lake allegations re. Windows 10
	 <p><i>Id.</i> at 11.</p>

96. Cypress Lake’s allegations regarding claim 1 are directed to basic Snap Assist features and functionality to resize side-by-side application windows. I understand that personal computing devices running Windows 10 with this functionality were in public use, on sale, and otherwise available to the public prior to the effective filing date. *See* Sundelin Decl. ¶¶ 19-25. It is my opinion, therefore, that prior art Windows 10 software and devices running Windows 10

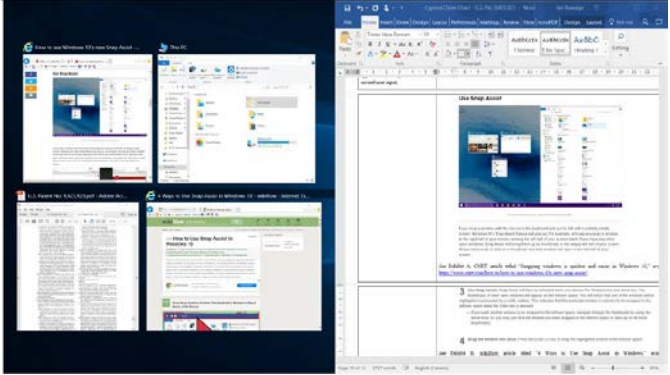
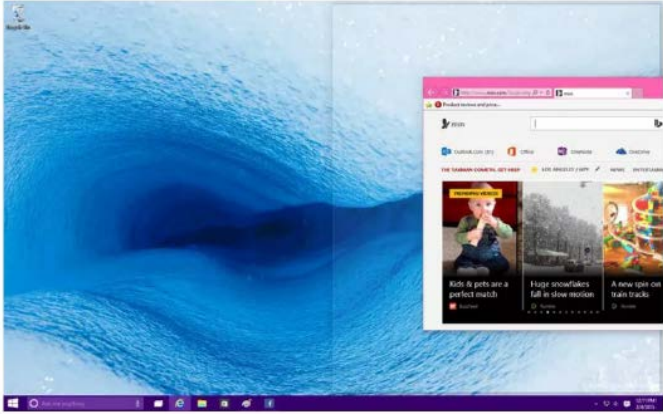
necessarily render claim 1 unpatentable under § 102(a)(1) or § 103 under Cypress Lake’s own interpretation of claim 47.

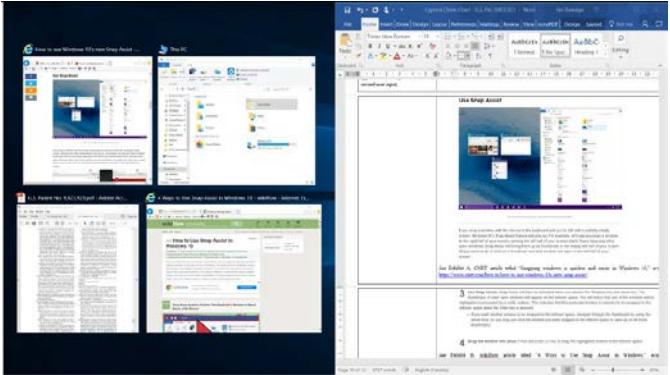
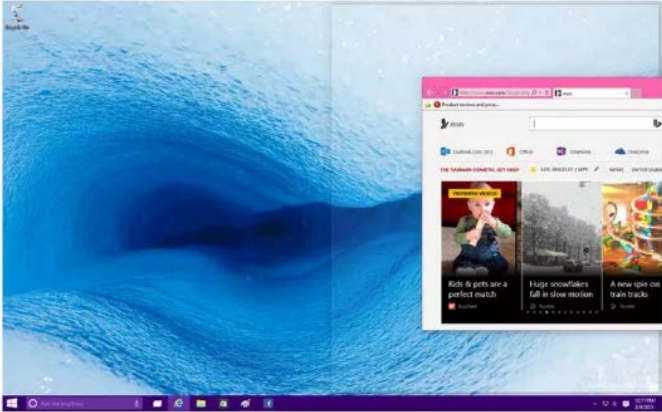
2. Dependent Claims 2, 4, 5, 6, 8, 10, and 11

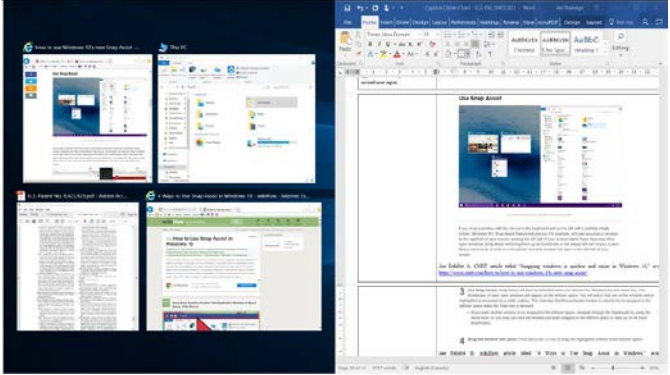
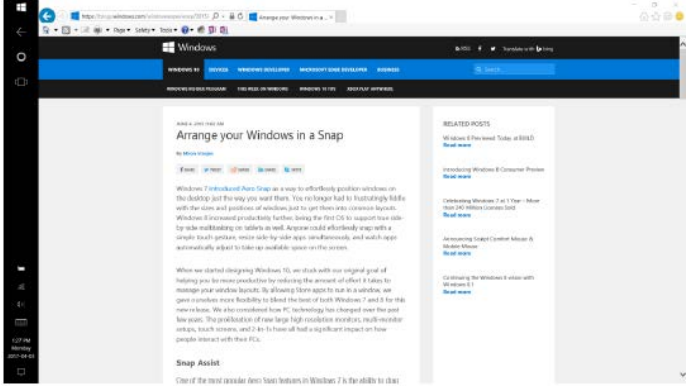
97. Cypress Lake similarly alleges that systems running Windows 10 meet each and every limitation of ’954 patent claims 2, 4, 5, 6, 8, 10, and 11. Again, the claims are not entitled to an earlier priority date, so under Cypress Lake’s own interpretation of this claim, each of the asserted claims would have been anticipated or rendered obvious by the Windows 10 devices already in public use, on sale, and otherwise available to the public.

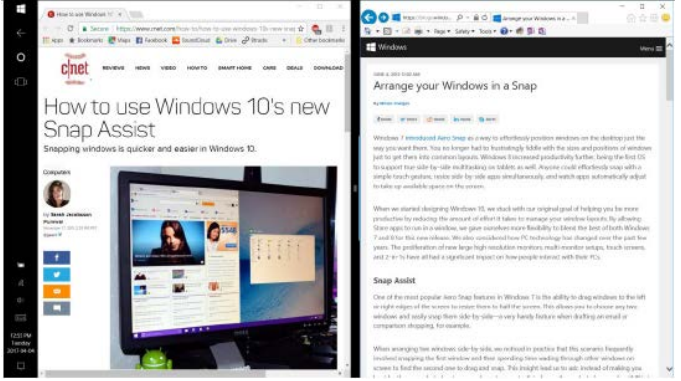
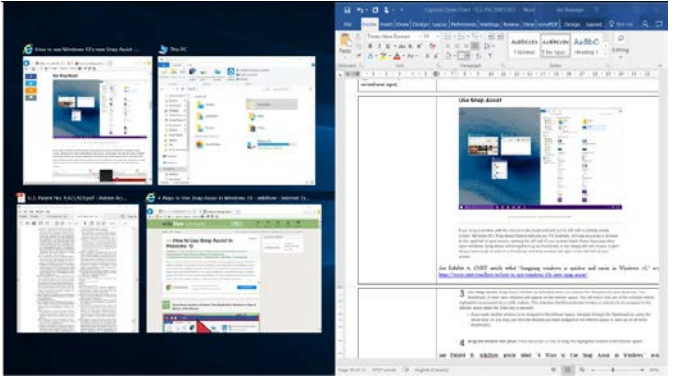
98. Cypress Lake’s claim charts provide the following infringement theories:

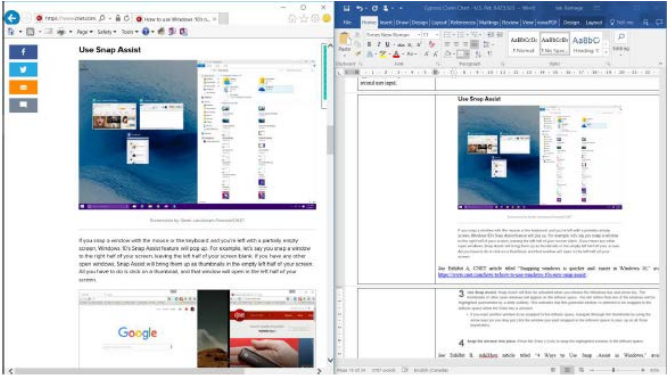
Claim	Cypress Lake allegations re. Windows 10
<i>2. The apparatus of claim 1, wherein the apparatus is configured such that the plurality of reduced windows are organized in at least one row of thumbnails that are presented in a desktop space exterior to the first window.</i>	Cypress Lake alleges that Windows 10 devices infringe this claim because “the SnapAssist menu consists of a row (or rows) of ‘thumbnails’ or reduced windows . . . and is displayed in an area of the screen outside the first ‘snapped’ window” Ex. 1009 at 12.

Claim	Cypress Lake allegations re. Windows 10
	 <p><i>Id.</i> at 13.</p>
<p>4. <i>The apparatus of claim 1, wherein the apparatus is configured such that the first user input includes a dragging and dropping in connection with the first window that results in a change in a size and a location of the first window.</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim because, when using SnapAssist with a “mouse, touchpad, or touchscreen,” “the ‘snapping’ of the first window involves dragging and dropping it . . . and doing so changes the window’s size and location . . .” <i>Id.</i> at 14.</p>  <p><i>Id.</i> at 15.</p>
<p>5. <i>The apparatus of claim 4, wherein the apparatus is configured such that at least a location of the plurality of reduced windows is based on the</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim because, “the Snap Assist menu is displayed opposite to where the window was snapped . . .” <i>Id.</i> at 16.</p>

Claim	Cypress Lake allegations re. Windows 10
<p><i>dragging and dropping in connection with the first window.</i></p>	 <p><i>Id.</i> at 18.</p>
<p><i>6. The apparatus of claim 1, wherein the apparatus is configured such that the first user input includes a dragging in connection with the first window that results in a change in a size and a location of the first window, where the first user input is capable of being received utilizing the at least one input device that includes a mouse pointer input device, and a touchscreen input device.</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim for the same reason it alleges they infringe claim 4. <i>Id.</i> at 14.</p>  <p><i>Id.</i> at 19.</p>
<p><i>8. The apparatus of claim 1, wherein the apparatus is configured such that the plurality of reduced windows including the second application reduced window associated with the second application and the third application reduced window</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim because, “the Snap Assist menu of thumbnails . . . is displayed beside the snapped application . . . and contains the thumbnails of the second and third applications only if they are active” <i>Id.</i> at 16.</p>

Claim	Cypress Lake allegations re. Windows 10
<p>associated with the third application, are selected for presentation adjacent to the first window associated with the first application based on whether the second application and the third application are operating.</p>	 <p><i>Id.</i> at 22.</p>
<p>10. The apparatus of claim 1, wherein the apparatus is configured such that the at least one input device and the display take the form of a touchscreen, and the first user input, the second user input, and the third user input include different touch inputs utilizing the touchscreen; and, until the first user input is detected, a maximum amount of a presentation space of the touchscreen capable of being used to present windows, is used for the first window, after which the presentation space of the touchscreen is split between two portions one of which is used for the first window.</p>	<p>Cypress Lake alleges that Windows 10 with touchscreens infringe this claim because a users could invoke SnapAssist using touch inputs and could maximize a single window to a full screen state prior to invoking SnapAssist. <i>Id.</i> at 22-24.</p>  <p><i>Id.</i> at 23 (depicting a maximized window)</p>

Claim	Cypress Lake allegations re. Windows 10
	 <p><i>Id.</i> at 24 (depicting side-by-side windows in Tablet Mode)</p>
<p>11. The apparatus of claim 1, wherein the apparatus is configured such that:</p> <p style="padding-left: 40px;"><i>the first user input includes a dragging in connection with the first window that results in a change in a size and a location of the first window;</i></p> <p style="padding-left: 40px;"><i>the plurality of reduced windows are spaced in available space of a desktop exterior to the first window at a location that is based on the dragging and dropping in connection with the first window; and</i></p> <p style="padding-left: 40px;"><i>the second application reduced window associated with the second application is a first sized representation that is enlarged to a second sized representation in connection with the presentation of the second window.</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim because:</p> <ul style="list-style-type: none"> • Snap Assist can be invoked by “dragging [a window] and changing its size and location,” • “the Snap Assist menu is presented in the half of the screen opposite to the snapped window,” and • the second window displayed using Snap Assist is “bigger than the thumbnail that was chosen” <p><i>Id.</i> at 22-24.</p> 

Claim	Cypress Lake allegations re. Windows 10
	 <p><i>Id.</i> at 25.</p>

99. As with claim 1, Cypress Lake’s allegations regarding these claims are directed to basic Snap Assist features and functionality to resize side-by-side application windows. As described above, I understand that personal computing devices running Windows 10 with this functionality were in public use, on sale, and otherwise available to the public prior to the effective filing date. *See Sundelin Decl.* ¶¶ 19-25. It is my opinion, therefore, that prior art Windows 10 software and devices running Windows 10 necessarily render each of these asserted claims unpatentable under § 102(a)(1) or § 103 under Cypress Lake’s own interpretation of the claim language.

3. Dependent Claims 12 and 13

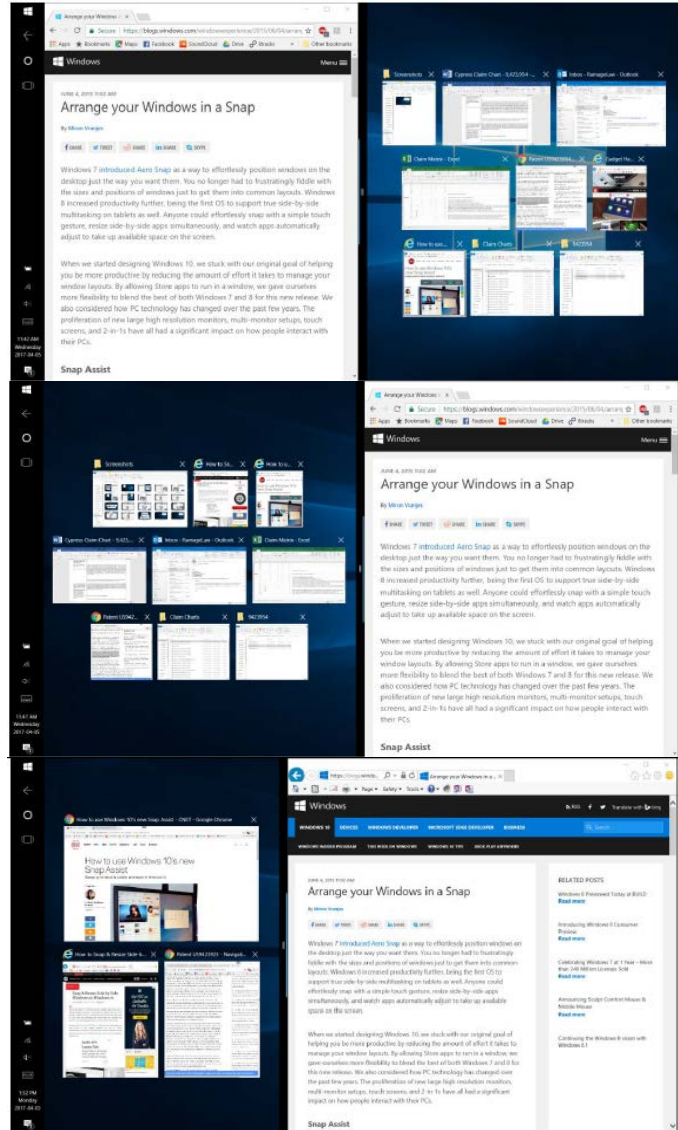
100. Cypress Lake similarly alleges that systems running Windows 10 meet each and every limitation of ’954 patent claims 12 and 13. Again, the claims are not entitled to an earlier priority date, so under Cypress Lake’s own

interpretation of this claim, each of the asserted claims would have been anticipated or rendered obvious by the Windows 10 devices already in public use, on sale, and otherwise available to the public.

101. Cypress Lake’s claim charts provide the following infringement theories:

Claim	Cypress Lake allegations re. Windows 10
<p>12(a).</p> <p><i>The apparatus of claim 1, wherein the apparatus is configured such that:</i></p> <p style="padding-left: 40px;"><i>the reduced windows include application window thumbnails that are presented in a desktop space exterior to the first window in response to the first user input which includes a dragging and dropping in connection with the first window that results in a change in a size and a location of the first window in connection with the presentation thereof;</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation for the same reasons Cypress Lake allege they infringe claims 2 and 4, above. Ex. 1009 at 26-27.</p>
<p>12(b)</p> <p><i>the application window thumbnails move in response to movement of the first window by a user, and are re-sized in response to re-sizing of the first window by the user, so that the</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation because “[w]hile the Snap Assist menu is displayed, a user can move it to the other side of the screen by swapping the location of the window” [and] re-size the menu via the vertical bar in the middle.” <i>Id.</i> at 27.</p>

application window thumbnails are located and sized to facilitate a selection thereof for user-configurable simultaneous presentation of multiple windows;

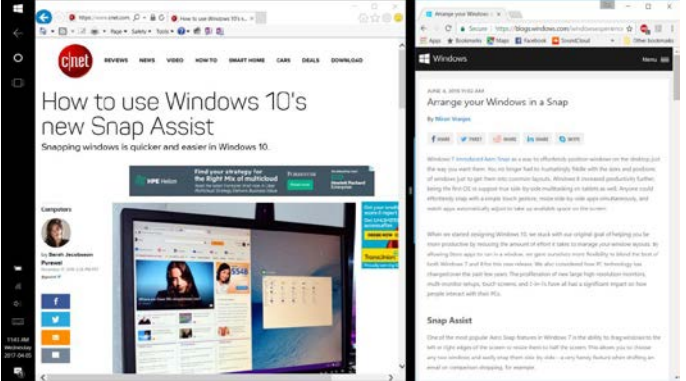


Id. at 26-29.

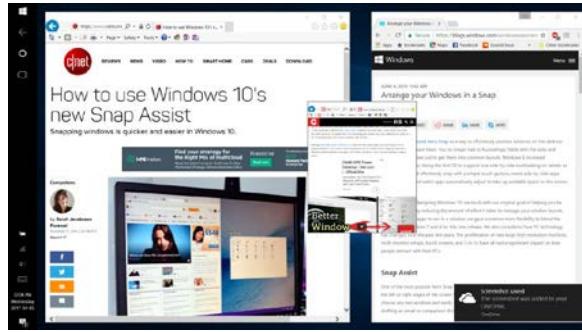
12(c)

the second user input includes a selection of the second application reduced window; the first application and the second application are operating before the first user input, the second user input, and

Cypress Lake alleges that Windows 10 devices satisfy this limitation because “choosing the second application involves selecting its thumbnail from the Snap Assist menu” and the windows displayed by Snap Assist belong to active windows. *Id.* at 29.

<p><i>the third user input are detected;</i></p>	
<p>12(d)</p> <p><i>the first size of the first window and the second size of the second window are symmetric in response to the second user input and asymmetric in response to the third user input, such that the first window and the second window are non-overlapping; and</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation because “after choosing the second window to display beside the first, the windows are the same size . . . ; after re-sizing them, they are different sizes . . . — and in neither case do they overlap.” <i>Id.</i> at 30-31.</p>  <p><i>Id.</i> at 31.</p>
<p>12(e)</p> <p><i>in response to a fourth user input, a third window is presented so as to overlie only some of the first window and the second window, such that:</i></p> <p><i>the third window has a first z-value that is different from a second z-value of the first window and the second window, in response to the fourth user input;</i></p> <p><i>the first size of the first window and the second size of the second window are changed, in response to the fourth user input; and</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation because “[t]he user may then select the Task View icon on the taskbar and choose a third application to display, and the Accused Windows Device will display that third window in reduced form, hovering over and in front of the first and second windows The addition of the third hovering window makes the other two windows slightly smaller . . . , and the third window initially moves slightly to the left, then to the right, and back to the center of the display, independent of any further user input and/or in anticipation of further user input, so as to indicate that further user input will have an effect on it” <i>Id.</i> at 30-31.</p>

the third window is given input focus, in response to the fourth user input.



Id. at 32.

Cypress Lake alleges the same functionality infringes claim 13:

13. The apparatus of claim 12, wherein the apparatus is configured such that the first z-value and the input focus of the third window facilitate user-configurable binding of the third window with respect to at least one of the first window or the second window.

Cypress Lake alleges that Windows 10 devices infringe this claim because “the hovering on top and animation of the third window” described for claim 12 “make it easier to choose whether to put the third window in place of the first or second window.” *Id.* at 34.

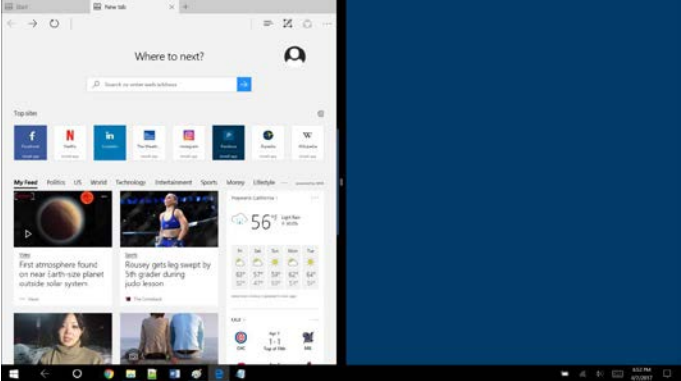
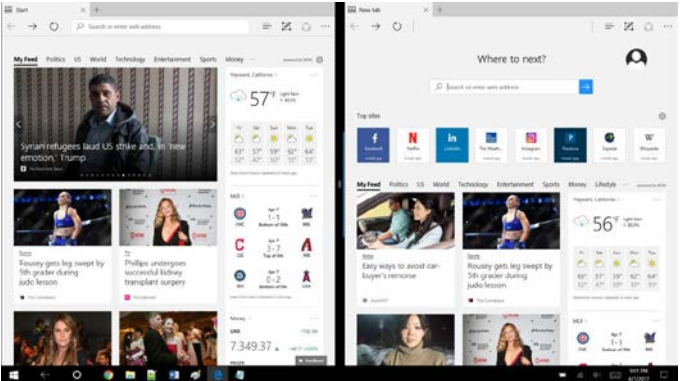
102. Cypress Lake’s allegations regarding these claims are directed to the interplay between the Snap Assist features and functionality for resizing Snap Assist in Tablet Mode via a shared divider and/or selecting a new window to replace side-by-side windows via the “teeter” user interface. I understand that personal computing devices running Windows 10 with this functionality were in public use, on sale, and otherwise available to the public prior to the effective filing date. *See Sundelin Decl.* ¶¶ 19-25. It is my opinion, therefore, that prior art

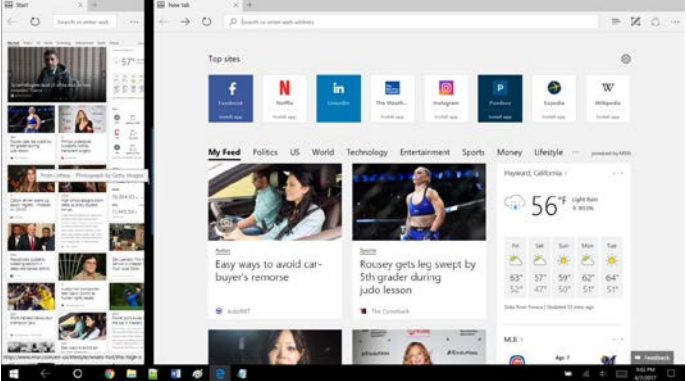
Windows 10 software and devices running Windows 10 necessarily render each of these asserted claims unpatentable under § 102.

4. Independent Claim 14

103. Cypress Lake similarly alleges that systems running Windows 10 meet each and every limitation of independent claim 14 of the '954 patent. Again, the claim is not entitled to an earlier priority date, so under Cypress Lake's own interpretation of this claim, each of the asserted claim would have been anticipated or rendered obvious by the Windows 10 devices already in public use, on sale, and otherwise available to the public.

Claim Limitation	Cypress Lake allegations re. Windows 10
14(a) <i>An apparatus, comprising: at least one processor configured for coupling with memory and a touchscreen, and further configured for:</i>	Cypress Lake has accused devices “running the Windows 10 Operating System” with “at least one processor” “configured to connect to a display,” “memory (RAM and hard drive,” and a “touchscreen.” Ex. 1009 at 34.
14(b) <i>storage of a plurality of applications including a first application, a second application, and a third application, utilizing the memory, the applications including a first program component and a second program component;</i>	Cypress Lake alleges that Windows 10 devices satisfy this limitation because they “can store three (or more) applications in . . . memory” and applications can include instances running “in separate tabs” (e.g., in the Microsoft Edge browser). <i>Id.</i> at 34.

Claim Limitation	Cypress Lake allegations re. Windows 10
<p>14(c)</p> <p><i>detection of a first user input;</i></p> <p><i>in response to the first user input, presentation of, utilizing the touchscreen, a first window associated with the first program component including at least one user interface element;</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation because they “can detect a user input via the touchscreen . . . to ‘snap’ an browser application window with tabs to either side of the screen” by dragging a window by its title bar to either side of the screen. <i>Id.</i> at 35.</p>  <p><i>Id.</i> at 35.</p>
<p>14(d)</p> <p><i>detection of a second user input in connection with the at least one user interface element of the first window;</i></p> <p><i>in response to the second user input in connection with the at least one user interface element of the first window, creation of a second window associated with the second program component and presentation thereof, utilizing the touchscreen, adjacent to and not overlapping with respect to the first window, for presenting,</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation because the user can “select and ‘pull’ the second tab out of the first window” to display in side-by-side with the first tab. <i>Id.</i> at 36.</p>  <p><i>Id.</i> at 37.</p>

Claim Limitation	Cypress Lake allegations re. Windows 10
<p><i>in the second window, data associated with the at least one user interface element of the first window;</i></p>	
<p>14(e) <i>detection of a third user input; and</i> <i>in response to the third user input, change, utilizing the touchscreen, the presentation of the first window and the second window, such that a first size of the first window and a second size of the second window are both changed, and the second window remains adjacent to and not overlapping with respect to the first window.</i></p>	<p>Cypress Lake alleges that Windows 10 devices satisfy this limitation by allowing the user to “select the vertical border between the two windows and drag it left or right to re-size the second window relative to the first” and the resizing the two windows. <i>Id.</i> at 37.</p>  <p><i>Id.</i> at 40.</p>

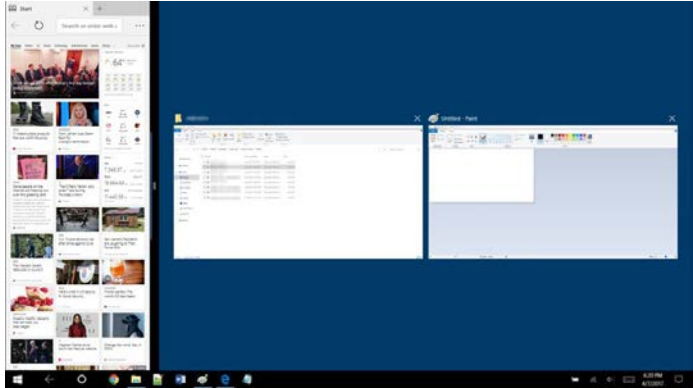
104. Cypress Lake’s allegations regarding these claims are directed to the interplay between the Snap Assist features and functionality for resizing side-by-side windows in Tablet Mode and/or dragging-and-dropping objects from a window to fill an empty space. I understand that personal computing devices running Windows 10 with this functionality were in public use, on sale, and otherwise available to the public prior to the effective filing date. *See Sundelin Decl.* ¶¶ 19-25. It is my opinion, therefore, that prior art Windows 10 software and

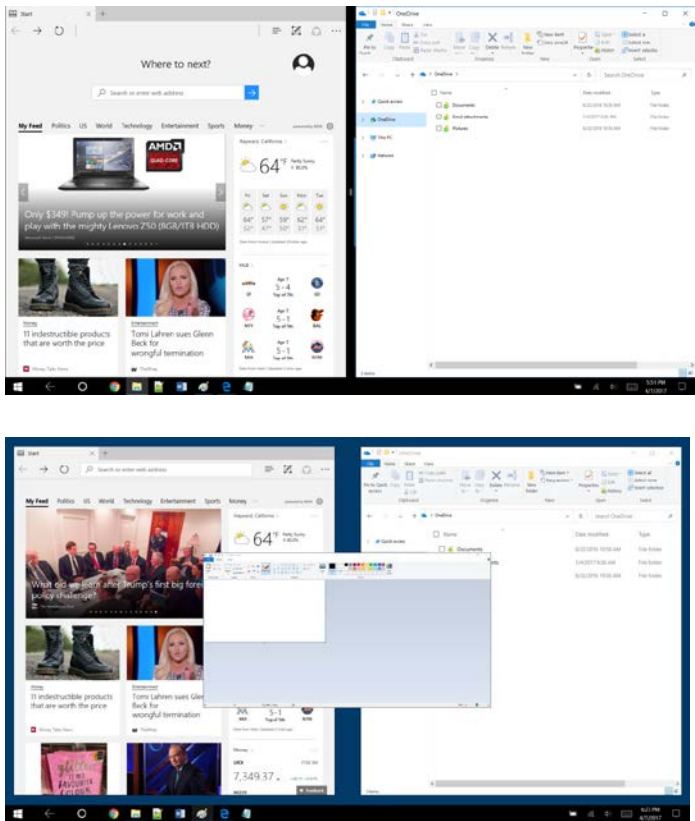
devices running Windows 10 necessarily render claim 14 unpatentable under § 102(a)(1) or § 103 under Cypress Lake’s own interpretation of the claim language.

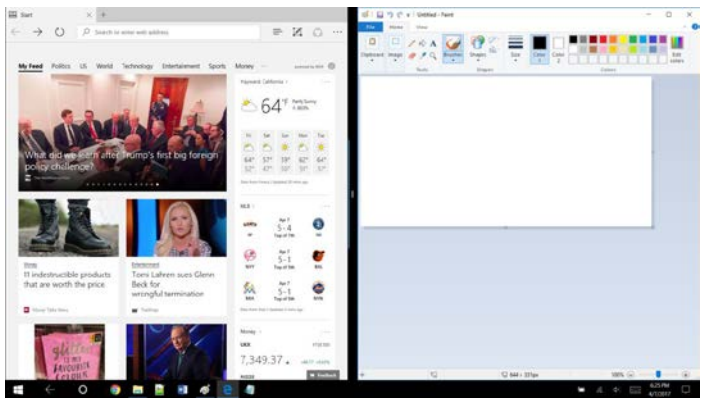
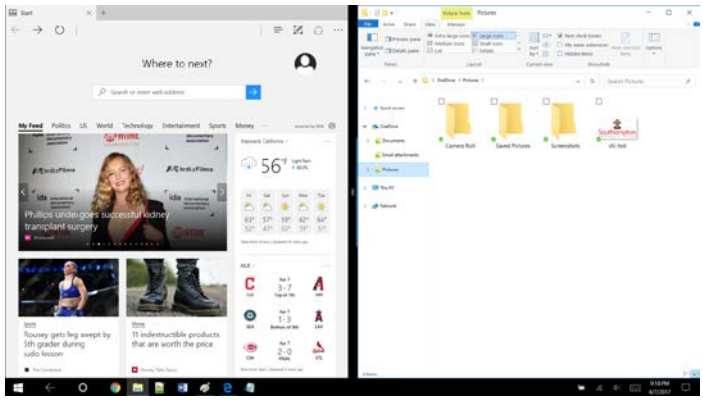
5. Dependent Claims 16, 17, and 18

105. Cypress Lake similarly alleges that systems running Windows 10 meet each and every limitation of ’954 patent claims 16, 17, and 18. Again, the claims are not entitled to an earlier priority date, so under Cypress Lake’s own interpretation of this claim, each of the asserted claims would have been anticipated or rendered obvious by the Windows 10 devices already in public use, on sale, and otherwise available to the public.

106. Cypress Lake’s claim charts provide the following infringement theories:

Claim	Cypress Lake allegations re. Windows 10
<p><i>16. The apparatus of claim 14, wherein the apparatus is configured such that the first window is presented simultaneously and without overlap with a plurality of reduced windows that are organized and do not overlap each other, the plurality of reduced windows including a second application reduced window associated with the second application and a third application reduced window associated with the third</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim because “if the user closes or minimizes the second window,” the SnapAssist menu reappears. Ex. 1009 at 40.</p> 

Claim	Cypress Lake allegations re. Windows 10
<p><i>application, for being selected for causing presentation via the second window in lieu of the data associated with the at least one user interface element of the first window.</i></p>	<p><i>Id.</i> at 41.</p>
<p><i>17. The apparatus of claim 14, wherein the apparatus is configured such that:</i></p> <p><i>in response to a fourth user input, a third window is presented so as to overlie only some of the first window and the second window, utilizing the touchscreen, such that:</i></p> <p><i>the third window has a first z-value that is different from a second z-value of the first window and the second window, in response to the fourth user input;</i></p> <p><i>the third window is given input focus, in response to the fourth user input; and</i></p> <p><i>the first size of the first window and the second size of the second window are changed, in response to the fourth user input; and</i></p> <p><i>in response to a fifth user input in the form of a dragging and dropping in connection with the third window, presentation of, utilizing the touchscreen, the third window associated with the third application in place of at</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim for the same reason Cypress Lake alleges they infringe claim 12 above, and because they further allow the user to drag the hovering window over either side-by-side window to replace it. <i>Id.</i> at 41-45.</p> 

Claim	Cypress Lake allegations re. Windows 10
<p><i>least one of the first window or the second window.</i></p>	 <p><i>Id.</i> at 44-45.</p>
<p><i>18. The apparatus of claim 17, wherein the apparatus is configured such that the fourth user input includes a selection of at least one other user interface element of the first window.</i></p>	<p>Cypress Lake alleges that Windows 10 devices infringe this claim because “the user also pull out GUI elements other than tabs—e.g., files out of File Explorer windows.” <i>Id.</i> at 45.</p>  <p><i>Id.</i> at 46.</p>

107. Cypress Lake’s allegations regarding these claims are directed to the interplay between the Snap Assist features and functionality for selecting a new window to replace side-by-side windows via the “teeter” user interface and/or dragging-and-dropping objects from a window to fill an empty space. I understand

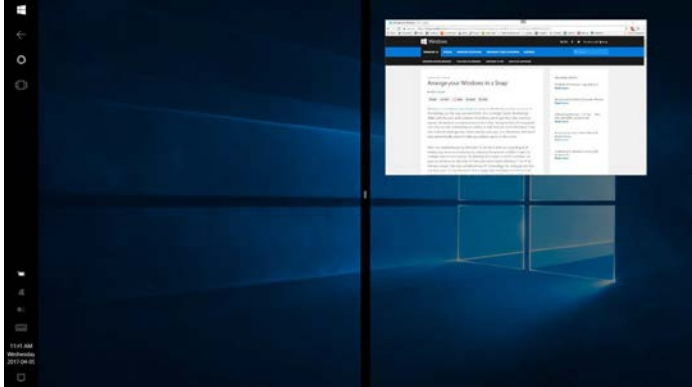
that personal computing devices running Windows 10 with this functionality were in public use, on sale, and otherwise available to the public prior to the effective filing date. *See* Sundelin Decl. ¶¶ 19-25. It is my opinion, therefore, that prior art Windows 10 software and devices running Windows 10 necessarily render each of these asserted claims unpatentable under § 102(a)(1) or § 103 under Cypress Lake’s own interpretation of the claim language.

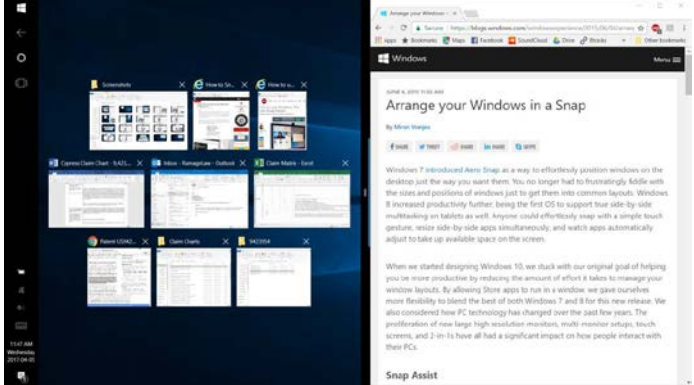
6. Claims 19 and 20

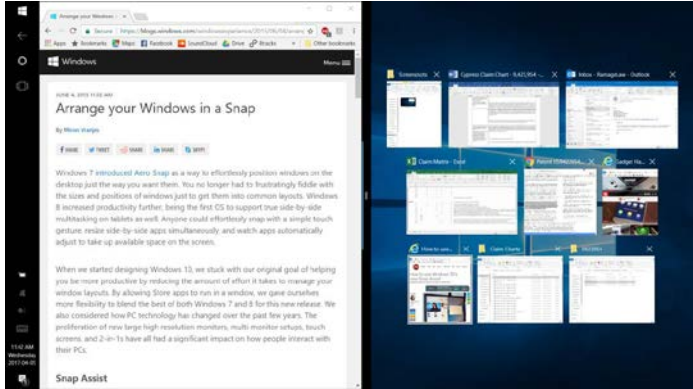
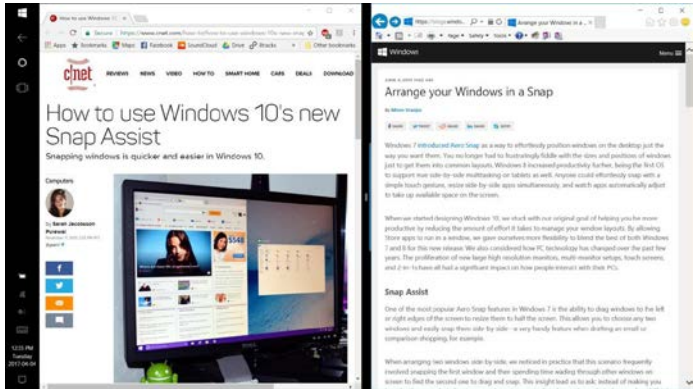
108. Cypress Lake similarly alleges that systems running Windows 10 meet each and every limitation of ’954 patent independent claim 19 and dependent claim 20. Again, the claims are not entitled to an earlier priority date, so under Cypress Lake’s own interpretation of this claim, each of the asserted claims would have been anticipated or rendered obvious by the Windows 10 devices already in public use, on sale, and otherwise available to the public.

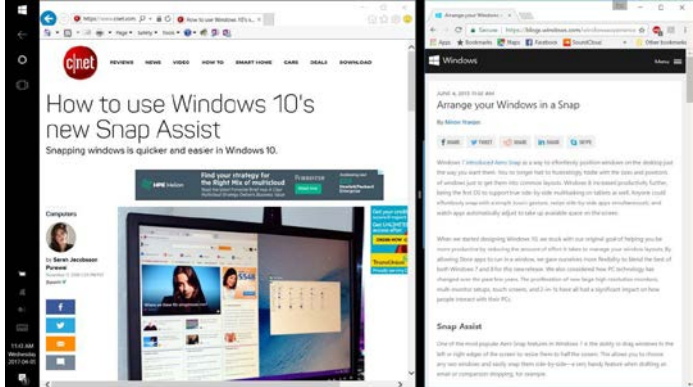
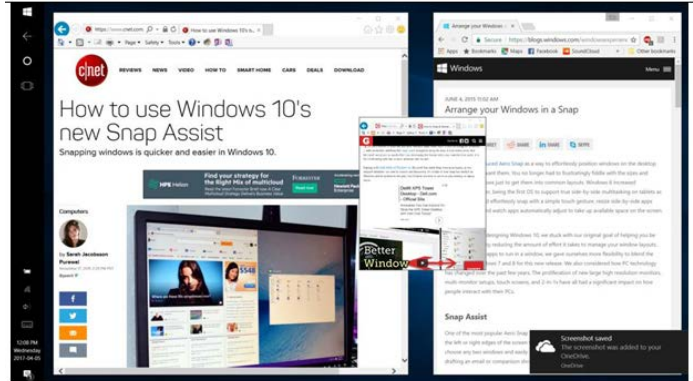
109. Cypress Lake’s claim charts provide the following infringement theories:

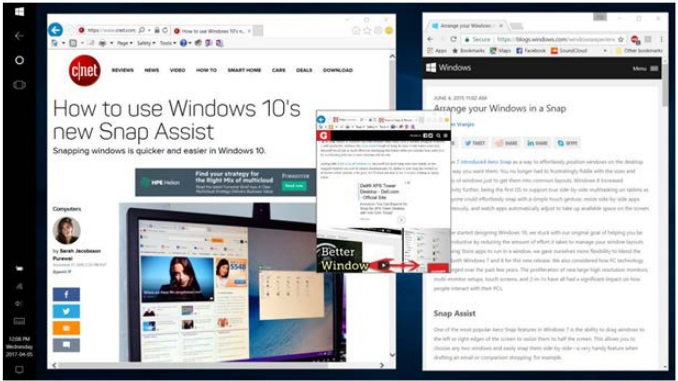
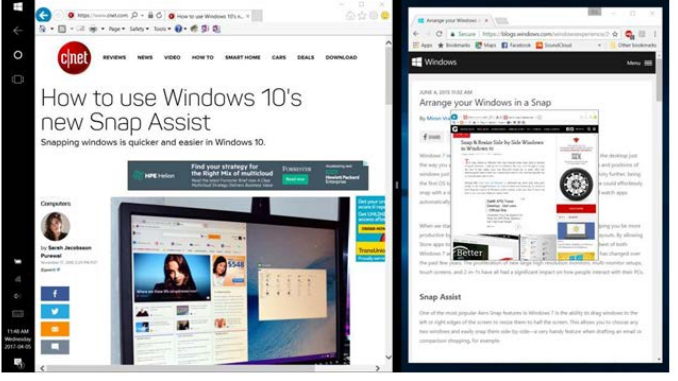
Claim	Cypress Lake allegations re. Windows 10
<p><i>19 (a)</i> <i>An apparatus, comprising: at least one processor configured for coupling with memory and a display and further configured for: storage of a first application, a</i></p>	<p>Cypress Lake accuses devices “running the Microsoft Windows 10 Operating System” with “at least one processor (<i>e.g.</i>, Intel Core i5), configured to connect memory (RAM and hard drive) and a display (<i>e.g.</i>, 14” LCD). . . .”). Ex. 1009 at 46.</p>

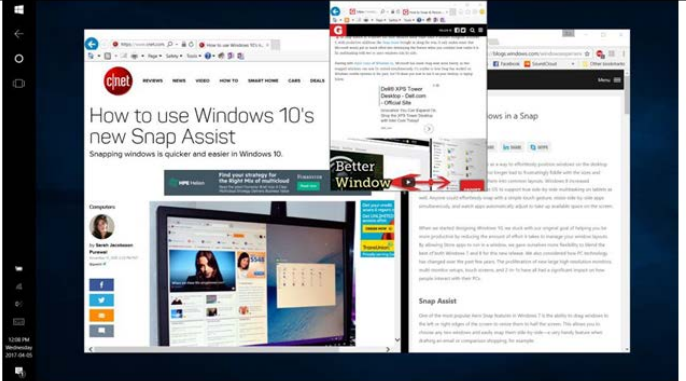

Claim	Cypress Lake allegations re. Windows 10
<p><i>second application, and a third application, utilizing the memory;</i></p>	
<p>19 (b)</p> <p><i>detection of a first user input in the form of a dragging and dropping;</i></p>	<p>Cypress Lake has alleges that Windows 10 devices satisfy this limitation because they “can sense when a user wants to . . . ‘snap’ a window . . . to either side of the screen. In Tablet mode, a user touches and pulls down the title bar at the top of a window, drags the window to the desired side, and releases. A user can also use the mouse, touchpad or other similar pointing device instead of the touchscreen, and can also select the window from the Task View menu instead of a maximized view.” <i>Id.</i> at 47.</p>  <p><i>Id.</i> at 48.</p>
<p>19 (c)</p> <p><i>in response to the first user input in the form of the dragging and dropping, presentation of, utilizing the display and in a desktop space, a plurality of window representations that are positioned in at least one row</i></p>	<p>Cypress Lake has alleged that Windows 10 devices satisfy this limitation because “after the ‘snapping’ of the first application window”, they “activate the ‘Snap Assist’ feature and display a menu of rows of thumbnails of two (or more) other open application windows . . . in an area of the screen opposite and not overlapping to where the user snapped the first window” <i>Id.</i> at</p>

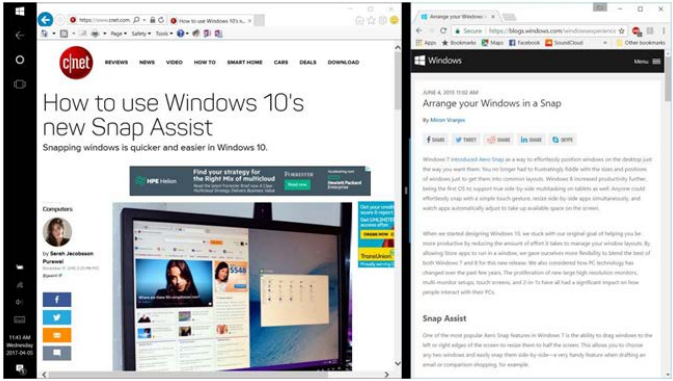
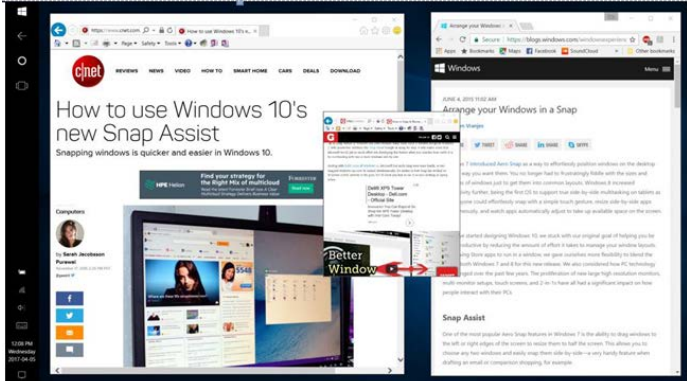
Claim	Cypress Lake allegations re. Windows 10
<p><i>and do not overlap each other, the plurality of window representations being associated with only operating applications except the first application, and including a second application window representation associated with the second application and a third application window representation associated with the third application both exterior to a first window associated with the first application that is presented without overlap between the first window and the plurality of window representations;</i></p>	<p>48.</p>  <p><i>Id.</i> at 49.</p>
<p>19 (d)</p> <p><i>detection of a second user input in the form of a dragging and dropping in connection with the first window associated with the first application;</i></p> <p><i>in response to the second user input in the form of the dragging and dropping in connection with the first window associated with the first application, changing both a first location of the first window and a second location of the plurality of window representations, such that the plurality of window representations do not overlap</i></p>	<p>Cypress Lake has alleged that Windows 10 devices satisfy this limitation because “in Tablet mode . . . , while the Snap Assist menu is displayed,” they can “detect a user input to swap the first window to the other side of the screen” and then “display the window on the left side . . . and the Snap Assist menu on the right, in rows, not overlapping the first window. . . .” <i>Id.</i> at 49-50.</p>

Claim	Cypress Lake allegations re. Windows 10
<p><i>each other and are positioned in at least one row exterior to the first window;</i></p>	 <p><i>Id.</i> at 50.</p>
<p>19 (e)</p> <p><i>detection of a third user input in connection with the second application window representation associated with the second application; in response to the third user input in connection with the second application window representation associated with the second application, presentation of, utilizing the display, a second window associated with the second application adjacent to the first window associated with the first application such that the first window and the second window do not overlap;</i></p>	<p>Cypress Lake has alleged that Windows 10 devices satisfy this limitation because they “can then detect (via touchscreen or mouse/touchpad) the user choosing the thumbnail of the second application window . . . , and display the second application window in the leftover space on the screen, without it touching the first window. . . .” <i>Id.</i> at 50.</p>  <p><i>Id.</i> at 51.</p>
<p>19 (f)</p> <p><i>detection of a fourth user input; in response to the fourth user input, changing the presentation</i></p>	<p>Cypress Lake has alleged that Windows 10 devices satisfy this limitation because they allow the user to “select the vertical border between the two windows and drag it left or right to re-size the second window relative to</p>

Claim	Cypress Lake allegations re. Windows 10
<p><i>of the first window and the second window, such that a first size of the first window and a second size of the second window are both changed;</i></p>	<p>the first” <i>Id.</i> at 51-52.</p>  <p><i>Id.</i> at 54.</p>
<p>19 (g)</p> <p><i>detection of a fifth user input; in response to the fifth user input, presentation of a third window so as to overlie only some of the first window and only some of the second window, utilizing the display, such that: the third window has a first z-value that is different from a second z-value of the first window and the second window, in response to the fifth user input, and the third window is given input focus, in response to the fifth user input;</i></p>	<p>Cypress Lake has alleged that Windows 10 devices satisfy this limitation because they allow the user to “select the Task View icon on the taskbar and choose a third application to display,” and then “will display that third window in reduced form, hovering over and in front of the first and second windows . . . and initially moving slightly to the left, then to the right, and back to the center of the display, independent of any further user input and/or in anticipation of further user input, so as to indicate that further user input will have an effect on it. . . .” <i>Id.</i> at 54.</p> 

Claim	Cypress Lake allegations re. Windows 10
	<p><i>Id.</i> at 55.</p>  <p><i>Id.</i> at 55.</p>
<p>19 (h)</p> <p><i>detection of a sixth user input in the form of a dragging and dropping in connection with the third window; and</i></p> <p><i>in response to the sixth user input in the form of the dragging and dropping in connection with the third window, presentation of, utilizing the display, the third window in place of at least one of the first window or the second window.</i></p>	<p>Cypress Lake has alleged that Windows 10 devices satisfy this limitation because they allow the user to “then drag and drop the third window onto either side (or top) of the display . . . [to] replace the first or second window (or both)” <i>Id.</i> at 56.</p>  <p><i>Id.</i> at 56.</p>

Claim	Cypress Lake allegations re. Windows 10
	 <p><i>Id.</i> at 57.</p>  <p><i>Id.</i> at 57.</p>
<p>20. <i>The apparatus of claim 19, wherein the apparatus is further configured such that the first size of the first window and the second size of the second window are changed, in response to the fifth user input.</i></p>	<p>Cypress Lake has alleged that Windows 10 devices infringe this claim because “in Tablet mode, when the third window is selected, the Accused Windows Device shrinks the first and second windows slightly. . . .” <i>Id.</i> at 58.</p>

Claim	Cypress Lake allegations re. Windows 10
	 <p><i>Id.</i> at 58.</p>  <p><i>Id.</i> at 59.</p>

110. Cypress Lake’s allegations regarding these claims are directed to the interplay between the Snap Assist and Task View features and functionality for resizing Snap Assist in Tablet Mode via a shared divider and/or selecting a new window to replace side-by-side windows via the “teeter” user interface. I understand that personal computing devices running Windows 10 with this functionality were in public use, on sale, and otherwise available to the public prior to the effective filing date. *See* Sundelin Decl. ¶¶ 19-25. It is my opinion,

therefore, that prior art Windows 10 software and devices running Windows 10 necessarily render each of these asserted claims unpatentable under § 102(a)(1) or § 103 under Cypress Lake's own interpretation of the claim language.

111. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; that these statements were made with knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001; and further that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

112. I declare under the penalty of perjury that the foregoing is true and correct.

//

//

//

//

//

//

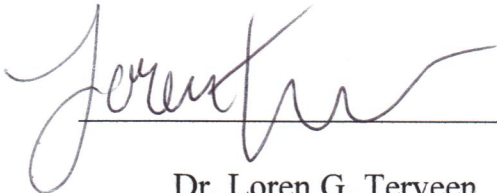
//

//

//

DECLARATION OF DR. LOREN G. TERVEEN
IN SUPPORT OF PETITION FOR POST-GRANT REVIEW
OF U.S. PATENT NO. 9,423,954

Executed on 23rd day of May 2017 in Minneapolis, MN.



Dr. Loren G. Terveen

Loren G. Terveen

terveen@cs.umn.edu	University of Minnesota	(612) 624-8310
--	-------------------------	----------------

Address Department of Computer Science and Engineering
 University of Minnesota
 Minneapolis, MN 55455

Education

Ph.D. in Computer Sciences	University of Texas	1991
M.S. in Computer Sciences	University of Texas	1988
B.A. in Computer Science, Mathematics, History	University of South Dakota	1984

Affiliations Member of ACM, ACM/SIGCHI

Research Interests Human-Computer Interaction, Computer-Supported Collaborative Work

Professional Experience

Professor	University of Minnesota	2002 --
Distinguished McKnight University Professor		2017--
Principal Member of Technical Staff	AT&T Labs - Research	1996-2002
Member of Technical Staff	AT&T Bell Labs	1991-1996
Graduate Research Intern	Microelectronics and Computer Corporation	1986-19991
Teaching Assistant	University of Texas at Austin	1985

AWARDS AND MAJOR ROLES

ACM Distinguished Scientist (awarded 2009)

ACM SIGCHI President (2015-2018)

ACM Council, SIG Governing Board Representative (2016 –)

RESEARCH

External Funding

NSF: “Computer-Supported Cooperative Work Doctoral Colloquium” **(PI)**, \$25,000 for 03/01/2016 to 02/28/2017.

National Institute on Drug Abuse: “A Technology-Delivered Peer-to-Peer Support ARB Adherence Intervention for HIV+ Adults”, (Co-PI, with Keith Horvath(PI) and Darin Erickson), \$3,302,62 for 07/01-2015 to 05/31/2020.

NSF: “HCC: Tools and Mechanisms to Support Social Participation Efforts”, **(PI)**, \$499,399 for 10/01/2012 to 09/30/2015.

NSF: “SoCS: Collaborative Research: Novel Algorithms and Interaction Mechanisms to Enhance Social Production”, **(PI)**, \$527,140, for 7/01/2012 to 06/30/2015 (recommended for funding).

NSF: “Collaborative Research: Supporting Newcomer Socialization in Online Production Communities”, **(PI)**, \$301,135.00 for 08/2011 to 08/31/2015.

Minnesota Department of Transportation: “Statewide Cycloplan: A Bicycle Planning Tool with Participatory GIS”, **(PI)**, \$130,000, for 10/01/2011 to 06/30/2013.

Metropolitan Council: “Cycloplan II”, **(PI)**, \$71,350, for 08/15/2011 to 05/31/2012.

NSF: “Wikisym Doctoral Consortium”, **(PI)**, \$13,163, for 05/01/2011 to 04/30/2012.

IBM: “Mobile Crowdsensing”, **(PI)**, \$100,000 awarded 04/01/2011

NSF: “Social-Computational Systems (SoCS) Community Meeting” **(PI)**, \$48,801 for 09/01/2010 to 08/31/2011

NSF: “SoCS: Collaborative Research: Information Framing: Intelligent Interfaces for an Online Production Community”, **(PI)**, \$375,000 for 09/15/2010 to 08/31/2015.

NSF: “Collaborative Research: Guiding Folksonomy Development to Enable Novel Tagging Applications” **(PI)**, with J. Riedl and S. Sen (Macalester College)). \$949, 788 for 04/1/2010 to 03/31/2014 (UMN Share).

Minnesota Department of Transportation: Bike, Bus, and Beyond: Extending Cyclopath to Enable MultiModal Routing **(PI)**, \$60,627, for 07/08/2010 to 01/31/2012

NIH: An Interactive Website to Promote Communication about Sexual Health and Dating

Relationships between Parents and Teens (**Co-PI**, with Sonya Brady (PI), Simon Rosser, and Renee Sieving), \$679,500 for 09/30/2009 to 08/31/2011.

Metropolitan Council: “Cycloplan” (**PI**), \$185,000, October 2009 – March 2011.

NSF “Collaborative Research: Understanding Online Volunteer Communities: Toward Theory-Based Design” (**co-PI**, with J. Riedl, J. Konstan, M. Snyder, & Y. Ren; R. Kraut (CMU) \$2,400,000 for 08/01/2008 to 07/31/2013.

NSF: “Recommender Systems Doctoral Consortium” (**PI**) \$15,415, 2007-2008.

NSF: “Mining Spatiotemporal Data: From Personal Use to Community Knowledge” (**PI**) \$449,570 for 12/1/2005 to 11/30/2009.

NSF: “Collaborative Research: Mark This! - Operationalizing the notion of "place" for interactive community systems” (**PI**, with Q. Jones (NJIT) and S. Whittaker (Univ. of Sheffield)). \$173,411 for 6/1/2003 to 5/31/2007 (UMN share).

NSF: “Being There: Mobile Devices for Community and Commerce” (**PI**, with J. Konstan, J. Riedl, and S. Shekhar). \$120,000 for 9/1/2002 to 8/31/2005.

NSF: “ITR: Collaborative Research: Designing On-Line Communities to Enhance Participation” (**co-PI**, with J. Konstan & J. Riedl, R. Kraut & S. Kiesler (CMU), P. Resnick and Y. Chen (Univ. of Michigan)). \$1,246,017 for 9/1/2003 to 8/31/2009 (UMN share).

AT&T: “VURI: Collaborative Filtering and Intelligent Interface Design for Enhanced TV Applications” (**PI**). \$35,000 for November 1, 2008 to October 31, 2009.

AT&T: “VURI: Collaborative Filtering and Intelligent Interface Design for Enhanced TV Applications” (**PI**). \$35,000 for June 1, 2007 to May 31, 2008.

Internal Funding

Minnesota/China Collaborative Research Grant: "Expertise Oriented Mining for Web Community". \$10,000 for July 1, 2007 to June 30, 2008 (PI, with Jie Tang, Tsinghua University, Beijing China).

University of Minnesota TEL grant: “The Next Generation Online Learning Environment: Designing for Community and Collaboration”. \$10,000 for September 2006 to May 2007. (co-PI, with Joan Hughes, David Ernst, and Ann Ooms, College of Education and Human Development).

University of Minnesota Digital Technology Center: “Indoor Navigation Aids for Visually Impaired People”. \$25,67 for June 2005 to December 2006. (co-PI, with S. Shekhar and G. Legge).

University of Minnesota Digital Technology Center: “Eye-Tracking Research on Community Websites: Photo Directories and Building Social Networks”. \$19,300 for June 2004 to June

2005. (co-PI, with J. Konstan).

University of Minnesota Grant-In-Aid: "Facilitating Participation in Online Communities".
\$20,397 for 1/1/2003 to 6/30/2004.

Books

1. Bickhard, M.H. and Terveen, L.G. Foundational Issues in Artificial Intelligence and Cognitive Science: Impasse and Solution, (1995), Elsevier Science.

Refereed Journal Papers

2. Thebault-Spieker, J., Terveen, L., and Hecht, B. Towards a Geographic Understanding of the Sharing economy: Systemic Biases in UberX and TaskRabbit. To Appear in *ACM Transactions on Computer-Human Interaction* (2017).
3. Brady, S.S., Sieving, R.E., Terveen, L.G., Rosser, R. S., Kodet, A.J., and Rothberg, V.D. An Interactive Website to Reduce Sexual Risk Behavior: Process Evaluation of TeensTalkHealth, *JMIR Research Protocols* (2015).
4. Ren, Y., Harper, F.M., Drenner, S., Terveen, L., Kiesler, S., Riedl, J., and Kraut, R.E. (2012). Building Member Attachment in Online Communities: Applying Theories of Group Identity and Interpersonal Bonds. *Management Information Systems Quarterly*.
5. Jones, Q., Grandhi, S., Karam, S., Whittaker, S., Zhou, C., and Terveen, L. Geographic 'Place' and Community Information Preferences, in *Computer-Supported Cooperative Work*.
6. Zhou, C., Frankowski, D., Ludford, P., Shekhar, S., Terveen, L., Discovering Personally Meaningful Places from Location Data: An Interactive Clustering Approach. *ACM Transactions on Information System*, 25, 3 (July 2007).
7. Ling, K., Beenen, G., Ludford, P.J., Wang, X., Chang, K., Li, X., Cosley, D., Frankowski, D., Terveen, L., Rashid, A.M., Resnick, P., and Kraut, R.E. Using Psychology to Motivate Contributions to Online Communities. *Journal of Computer-Mediated Communication*, 10, 4 (June 2005).
8. Terveen, L. and McDonald, D. Social Matching: A Framework and Research Agenda. *ACM Transactions on Computer-Human Interaction*, 12, 3 (2005), 401-434.
9. Whittaker, S., Jones, Q., Nardi, B., Creech, M., Terveen, L., Isaacs, E., and Hainsworth, J. ContactMap: organizing communication in a social desktop, in *ACM Transactions on Computer-Human Interaction*, 11, 4 (December 2004), 445-471.
10. Jones, Q., Grandhi, S., Terveen, L., and Whittaker, S. People-To-People-to-Geographical-Places: The P3 Framework for Location-Based Community Systems. in *Computer-Supported Cooperative Work*, 13, 3-4 (August 2004), 249-282.
11. Herlocker, J.L., Konstan, J.A., Terveen, L.G., and Riedl, J.T. Evaluating Collaborative Filtering Recommender Systems, *ACM Transactions on Information Systems* (2004).
12. Amento, B., Terveen, L., Hill, W., Hix, D., and Schulman, R. Experiments in Social Data Mining: The TopicShop System, in *ACM Transactions on Computer-Human Interaction*, 10, 1 (March 2003), 54-85.
13. Whittaker, S., Terveen, L.G, and Nardi, B.A. Let's stop pushing the envelope and start addressing it, in *Human-Computer Interaction*, 15, 2-3 (Sep 2000), 75-106.
14. Terveen, L.G., Hill, W.C., and Amento, B. Constructing, Organizing, and Visualizing Collections of Topically Related Web Resources, in *ACM Transactions on Computer-Human Interaction*, 6, 1 (Mar. 1999), 67-94.

15. Selfridge, P.G. and Terveen, L.G. Knowledge Management Tools for Business Process Support and Reengineering, in *Journal of Intelligent Systems in Accounting, Finance, and Management* (Jan. 1996).
16. Terveen, L.G. An Overview of Human-Computer Collaboration, in *Knowledge-Based Systems*, 8, 2-3 (1995), 67-81.
17. Terveen, L.G., Selfridge, P.G., and Long, M.D. Living Design Memory: Framework, System, and Lessons Learned, in *Human-Computer Interaction*, 10, 1 (1995), 1-37.
18. Terveen, L.G. Intelligent Systems as Cooperative Systems, in *International Journal of Intelligent Systems*, 3, 2-4 (1993), 217-250.
19. Brachman, R.J., Selfridge, P.G., Terveen, L.G., Altman, B., Borgida, A., Halper, F., Kirk, T., Lazar, A., McGuinness, D.L., and Resnick, L.A. Integrated Support for Data Archaeology, in *International Journal of Intelligent and Cooperative Information Systems*, 2, 2 (1993), 159-185.

Refereed Conference Papers

20. Miller, H., Kluver, D., Thebault-Spieker, J., Terveen, L., and Hecht, B. Understanding Emoji Ambiguity in Context: The Role of Text in Emoji-Related Miscommunication. In *Proceedings of ICWSM 2017, AAAI Conference on the Web and Social Media*.
21. Hall, A., McRoberts, S., Thebault-Spieker, J., Lin, A.Y., Sen, S., Hecht, B., and Terveen, L. Freedom versus Standardization: Structured Data Generation in a Peer Production Community. In *Proceedings of CHI 2017, the ACM Conference on Human Factors in Computing Systems (CHI 2017)*.
22. Yu, B., Ren, Y., Terveen, L., and Zhu, H. Predicting Member Productivity and Withdrawal from Pre-Joining Attachments in Online Production Groups. In *Proceedings of CSCW 2017, The ACM Conference on Computer-Supported Cooperative Work and Social Computing*.
23. Chang, S., Harper, M., and Terveen, L. Crowd-Based Personalized Natural Language Explanations for Recommendations In *Proceedings of RecSys 2016, The ACM Conference on Recommender Systems*.
24. Chang, S., Harper, M., He, L., and Terveen, L. CrowdLens: Experimenting with Crowd-Powered Recommendation and Explanation. In *Proceedings of ICWSM 2016, AAAI Conference on the Web and Social Media*.
25. Miller, H., Thebault-Spieker, J., Chang, S., Johnson, I., Terveen, L., and Hecht, B. “Blissfully happy” or “ready to fight”: Varying Interpretations of Emoji. In *Proceedings of ICWSM 2016, AAAI Conference Web and Social Media*.
26. Zhao, Q., Huang, Z., Harper, F.M., Terveen, L., and Konstan, J. “Precision Crowdsourcing: Closing the Loop to Turn Information Consumers into Information Producers”. In *Proceedings of CSCW 2016, The ACM Conference on Computer Supported Cooperative Work and Social Computing*.
27. Harper, F.M., Xu, F., Kaur, H., Condiff, K., Chang, S., and Terveen, L. Putting Users in Control of their Recommendations. In *Proceedings of RecSys2015, The ACM Conference on Recommender Systems*.
28. Kapoor, K., Kumar, V., Terveen, L., Konstan, J.A., and Schrater, P. “I like to explore sometimes” – Adapting to Dynamic User Novelty Preferences. In *Proceedings of RecSys2015, The ACM Conference on Recommender Systems*.

29. Warncke-Wang, M., Ranjan, V., Terveen, L., and Hecht, B. Misalignment Between Supply and Demand of Quality Content in Peer Production Communities. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM'15)*.
30. Chang, S., Harper, F.M., and Terveen, L. Using Groups of Items to Bootstrap New Users in Recommender Systems. In *Proceedings of the 18th ACM conference on Computer supported cooperative work & social computing (CSCW '15)*. ACM, New York, NY, USA.
31. Miller, H., Chang, S., and Terveen, L. "I LOVE THIS SITE!" vs. "It's a little girly": Perceptions of and Initial User Experience with Pinterest. In *Proceedings of the 18th ACM conference on Computer supported cooperative work & social computing (CSCW '15)*. ACM, New York, NY, USA.
32. #Thebault-Spieker, J., Terveen, L., and Hecht, B. 2015. Avoiding the South Side and the Suburbs: The Geography of Mobile Crowdsourcing Markets. In *Proceedings of the 18th ACM conference on Computer supported cooperative work & social computing (CSCW '15)*. ACM, New York, NY, USA.
33. Warncke-Wang, M., Hecht, B., and Terveen, L. The Success and Failure of Quality Improvement Projects in Peer Production Communities. In *Proceedings of the 18th ACM conference on Computer supported cooperative work & social computing (CSCW '15)*. ACM, New York, NY, USA.
34. Panciera, K., Masli, M., and Terveen, L. Crème de la Crème: Elite contributors in an online community, in *Proceedings of the 10th International Symposium on Open Collaboration (OpenSym 2014)*.
35. Kumar, V., Kluver, D., Terveen, L., and Riedl, J. More Efficient Tagging Systems with Tag Seeding, in *Proceedings of IEEE SocialCom 2014*.
36. Nguyen, T., Hui, P., Harper, F.M., Terveen, L., and Konstan, J. Exploring the Filter Bubble: The Effect of Using Recommender Systems on Content Diversity, in *Proceedings of WWW 2014*.
37. Halfaker, A., Geiger, S., and Terveen, L. Snuggle: Designing for Efficient Socialization and Ideological Critique, in *Proceedings of CHI 2014*.
38. Chang, S., Kumar, V., Gilbert, E., and Terveen, L. Specialization, Homophily, and Gender in a Social Curation Site Findings from Pinterest, in *Proceedings of CSCW 2014*.
39. Grevet, C., Terveen, L., and Gilbert, E. Managing Political Differences in Social Media, in *Proceedings of CSCW 2014*.
40. Masli, M., and Terveen, L. Leveraging the Contributory Potential of User Feedback, in *Proceedings of CSCW 2014*.
41. Sheppard, S.A., Wiggins, A., and Terveen, L. Capturing Quality: Retaining Provenance for Curated Volunteer Monitoring Data, in *Proceedings of CSCW 2014*.
42. DeLong, C., Terveen, L., and Srivastava, J. (2013). TeamSkill and the NBA: Applying Lessons from the Virtual World to the Real World, in *Proceedings of ASONAM 2013*.
43. Torre, F., Liu, Y., Liu, Z., and Terveen, L. (2013). Local Knowledge Matters for Crowdsourcing Systems: Experience from Transferring an American Site to China, in *Proceedings of ICWSM 2013*.
44. Gilbert, E., Bakshi, S, Chang, S., and Terveen, L. (2013). "I Need to Try This!": A Statistical Overview of Pinterest, in *Proceedings of CHI 2013*.
45. Torre, F., Pitchford, D., Brown, P., and Terveen, L. (2012). Matching GPS Traces to (Possibly) Incomplete Map Data: Bridging Map Building and Map Matching, in *ACM SIGSPATIAL GIS 2012*.

46. Dunne, L., Zhang, J., and Terveen, L. (2012). An Investigation of Contents and Use of the Home Wardrobe, in UbiComp 2012.
47. Dong, Z., Shi, C., Sen, S., Terveen, L., and Riedl, J. (2012). War Versus Inspirational in Forrest Gump: Cultural Effects in Tagging Communities, in ICWSM 2012.
48. Masli, M. and Terveen, L. (2012) Evaluating Compliance-Without-Pressure Techniques for Increasing Participation in Online Communities, in CHI 2012.
49. Fuglestad P.T., Dwyer, P.C., Filson Moses, J., Kim, J.S., Mannino, C.A., Terveen, L., and Snyder, M. (2012) What Makes Users Rate (Share, Tag, Edit...)? Predicting Patterns of Participation in Online Communities, in CSCW 2012.
50. Nathan, M., Topkara, M., Lai, J., Pan, S., Wood, S., Boston, J., and Terveen, L. (2012) In Case You Missed It: Benefits of Attendee-Shared Annotations for Non-attendees of Remote Meetings, in CSCW 2012.
51. Priedhorsky, R., Pitchford, D., Sen, S., and Terveen, L. (2012), Recommending Routes in the Context of Bicycling: Algorithms, Evaluation, and the Value of Personalization, in CSCW 2012.
52. Lam, S.K., Uduwage, A., Dong, Z., Sen, S., Musicant, D.R., Terveen, L., and Riedl, J. (2011). WP:Clubhouse? An Exploration of Wikipedia's Gender Imbalance, in Wikisym 2011. **Best Paper Winner.**
53. Panciera, K., Masli, M., and Terveen, L.G. (2011). "How Should I Go from __ to __ without Getting Killed? Motivations and Benefits in Open Collaboration", in Wikisym 2011.
54. Priedhorsky, R., and Terveen, L.G. (2011). Wiki Grows Up: Arbitrary Data Models, Access Control, and Beyond, in Wikisym 2011.
55. Sheppard, S.A. and Terveen, L.G. (2011). Quality is a Verb: The Operationalization of Data Quality in a Citizen Science Community, in Wikisym 2011.
56. Masli, M., Priedhorsky, R., and Terveen, L. (2011). Task Specialization in Social Production Communities: The Case of Geographic Volunteer Work, in the Proceedings the 4th International AAAI Conference on Weblogs and Social Media (ICWSM 2011).
57. Torre, F., Sheppard, S.A., Priedhorsky, R., and Terveen, L. (2010) bumpy, caution with merging: An Exploration of Tagging in a Geowiki, in GROUP 2010.
58. Panciera, K., Priedhorsky, R., Erickson, T., and Terveen, L. (2010). Lurking? Cyclopaths? A Quantitative Analysis of User Behavior in a Geowiki, in CHI 2010. **Best of CHI Nominee.**
59. Priedhorsky, R., Masli, M., and Terveen, L. (2010). Eliciting and Focusing Geographic Volunteer Work, in CSCW 2010.
60. Panciera, K., Halfaker, A., and Terveen, L. (2009). Wikipedians are Born, Not Made: A Study of Power Editors on Wikipedia, in GROUP 2009. *36% Acceptance Rate.*
61. Reily, K., Ludford Finnerty, P., and Terveen, L. Two Peers are Better than One: Aggregating Peer Reviews for Computing Assignments is Surprisingly Accurate, in GROUP 2009. *36% Acceptance Rate.*
62. Ludwig, M., Priedhorsky, R., and Terveen, L. (2009). Path Selection: A Novel Interaction Technique for Mapping Applications, to appear in CHI 2009. *24% acceptance rate.*
63. Priedhorsky, R., and Terveen, L. (2008). The Computational Geowiki: What, Why, and How, in CSCW 2008. *23% acceptance rate.* **Best of CSCW Nominee.**
64. Drenner, S., Sen, S., and Terveen, L (2008). Crafting the Initial User Experience to Achieve Community Goals, in RecSys 2008. *30% acceptance rate.*
65. Reily, K., Ludford, P., and Terveen, L. (2008). Sharescape: An Interface for Place Annotation, in NordiCHI 2008. *30% acceptance rate.*

66. Nathan, M., Harrison, C., Yarosh, S., Terveen, L., Stead, L., and Amento, B. (2008), CollaboraTV: Making Television Viewing Social Again, in uxTV 2008.
67. Priedhorsky, R., Jordan, B., and Terveen, L. (2007), How a Personalized Geowiki Can Help Bicyclists Share Information More Effectively, in WikiSym 2007. 50% acceptance rate.
68. Priedhorsky, R., Chen, J., Lam, A., Panciera, K., Terveen, L., and Riedl, J. (2007), Creating, Destroying, and Restoring Value in Wikipedia, in Group 2007.
69. Rouben, A. and Terveen, L. (2007), Speech and Non-Speech Audio: Navigational Information and Cognitive Load, in International Conference on Auditory Displays (ICAD).
70. Ludford, P., Priedhorsky, R., Reily, K., and Terveen, L. (2007), Capturing, Sharing, and Using Local Place Information, in CHI 2007. 25% acceptance rate.
71. Cosley, D., Frankowski, D., Terveen, L., and Riedl, J. (2007), SuggestBot: Using Intelligent Task Routing to Help People Find Work in Wikipedia, in IUI 2007. 22% acceptance rate.
72. Harper, F.M., Frankowski, D., Drenner, S., Ren, Y., Kiesler, S., Terveen, L., Kraut, R., and Riedl, J. (2007), Talk Amongst Yourselves: Inviting Users To Participate In Online Conversations, in IUI 2007. 22% acceptance rate.
73. Frankowski, D., Cosley, D., Sen, S., Terveen, L., and Riedl, J. You Are What You Say: Privacy Risks of Public Mentions, in SIGIR 2006. 19% acceptance rate.
74. Ludford, P.J., Frankowski, D., Reily, K., Wilms, K., and Terveen, L., Because I Carry My Cell Phone Anyway: Functional Location-Based Reminder Applications, in *Proceedings of CHI 2006*. 23% acceptance rate.
75. Cosley, D., Frankowski, D., Terveen, L., and Riedl, J., Using Intelligent Task Routing and Contribution Review to Help Communities Build Artifacts of Lasting Value, in *Proceedings of CHI 2006*. 23% acceptance rate.
76. Drenner, S., Harper, M., Frankowski, D., Riedl, J., and Terveen, L. Insert Movie Reference Here: A System to Bridge Conversation and ItemOriented Web Sites, in *Proceedings of CHI 2006* (Tech Note).
77. Zhou, C., Ludford, P., Frankowski, D., and Terveen, L. How Do People's Concepts of Place Relate to Physical Locations? In *Proceedings of INTERACT 2005*. 27% acceptance rate.
78. D. Cosley, D. Frankowski, S. Kiesler, L. Terveen, J. Riedl. How Oversight Improves Member-Maintained Communities. In *Proceedings of CHI 2005*, Portland, OR, 2005. 25% acceptance rate.
79. Jones, Q., Grandhi, S., Whittaker, S., Chivakula, K., and Terveen, L. Putting Systems into Place: A Qualitative Study of Design Requirements for Location Aware Community Systems, in *Proceedings of CSCW 2004*. 30% acceptance rate.
80. Zhou, C., Ludford, P., Shekhar, S., and Terveen, L. Discovering Personal Gazetteers: An Interactive Clustering Approach, in *ACM GIS 2004* (12th International Symposium on Geographic Information Systems). 33% acceptance rate.
81. Ludford, P., Cosley, D., Frankowski, D., and Terveen, L.G. Think Different: Increasing Online Community Participation Using Uniqueness and Group Dissimilarity, in *Proceedings of CHI 2004*. 16% acceptance rate.
82. Cosley, D., Ludford, P. and Terveen, L.G. Studying the Effect of Similarity in Online Task-Focused Interactions, *Proceedings of GROUP 2003*. 35% acceptance rate.
83. Ludford, P. and Terveen, L.G. Does an Individual's Myers-Briggs Type Indicator Preference Influence Task-Oriented Technology Use?, *Proceedings of Interact 2003*. 34% acceptance rate.

84. Whittaker, S., Jones, Q., and Terveen, L.G. Contact Management: Identifying Contacts to Support Long-Term Communication, *Proceedings of CSCW 2002*, 216-225. 20% acceptance rate.
85. Terveen, L.G., McMackin, J., Amento, B., and Hill, W. Specifying Preferences Based On User History, *Proceedings of CHI 2002*, 315-322. 15% acceptance rate.
86. Whittaker, S., Jones, Q., and Terveen, L.G. Managing Long Term Conversations: Conversation and Contact Management, *Proceedings of HICSS 2002*. 50% acceptance rate.
87. Amento, B., Terveen, L., Hill, W., and Hix, D. TopicShop: Enhanced Support for Evaluating and Organizing Collections of Web Sites, *Proceedings of UIST 2000*. 26% acceptance rate.
88. Amento, B., Terveen, L., and Hill, W. Does 'Authority' Mean Quality? Predicting Expert Quality Ratings of Web Documents, *Proceedings of SIGIR 2000*. 27% acceptance rate.
89. Amento, B., Hill, W., Terveen, L., Hix, D., and Ju, P. An Empirical Evaluation of User Interfaces for Topic Management of Web Sites, *Proceedings of CHI 1999*, 552-559. 25% acceptance rate.
90. Terveen, L.G and Hill, W.C. Evaluating Emergent Collaboration on the Web, *Proceedings of CSCW 1998*, 355-362. 19% acceptance rate.
91. Whittaker, S., Terveen, L.G, Hill, W.C., and Cherny, L. The Dynamics of Mass Interaction, *Proceedings of CSCW 1998*, 257-264. 19% acceptance rate.
92. Terveen, L.G and Hill, W.C. Finding and Visualizing Inter-site Clan Graphs, *Proceedings of CHI 1998*, 448-455. 23% acceptance rate.
93. Terveen, L.G and Hill, W.C. Involving Users in Continuous Design of Web Content, in *Proceedings of DIS 1997*, ACM Press, 137-145. 41% acceptance rate
94. Terveen, L.G., Hill, W.C., Amento, B., McDonald, D., and Creter, J. Building Task-Specific Interfaces to High Volume Conversational Data, *Proceedings of CHI 1997*, 226-233. 23% acceptance rate.
95. Hill, W.C. and Terveen, L.G Using Frequency-of-Mention in Public Conversations for Social Filtering, *Proceedings of CSCW 1996*, 106-112.
96. Terveen, L.G. and Murray, L. Helping Users Program Their Personal Agents, *Proceedings of CHI 1996*, 355-361. 23% acceptance rate.
97. Terveen, L.G. and Tuomenoksa, M.L. DynaDesigner: A Tool for Rapid Creation of Device-Independent, *Proceedings of INTERACT 1995*, 386-389.
98. Terveen, L.G. and Selfridge, P.G. Intelligent Assistance for Software Construction: A Case Study, *Proceedings of Knowledge-Based Software Engineering 1994*, 14-21.
99. Terveen, L.G., Selfridge, P.G., and Long, M.D. From 'Folklore' to 'Living Design Memory', in *Proceedings of INTERCHI 1993*, 15-22. 19% Acceptance Rate.
100. Terveen, L.G. Interface Support for Data Archaeology, *ISMM International Conference on Information and Knowledge Management (CIKM'93)*. (acceptance rate for 1992 unknown; average is 24%).
101. Brachman, R. J., Selfridge, P.G., Terveen, L.G., Altman, B., Borgida, A., Helper, F., Kirk, T., Lazar, A., McGuinness, D.L., and Resnick, L. A., Knowledge Representation Support for Data Archaeology, *ISMM International Conference on Information and Knowledge Management (CIKM'92)*. (acceptance rate for 1992 unknown; average is 24%).
102. Selfridge, P.G., Terveen, L.G., and Long, M.D. Managing Design Knowledge to Provide Assistance to Large-Scale Software Development, *Proceedings of Knowledge-Based Software Engineering 1992*, 154-162.

103. Terveen, L.G. and Wroblewski, D.A. A Tool for Achieving Consensus in Knowledge Editing, *Proceedings of AAAI 1991*, 74-79. 24% acceptance rate.
104. Terveen, L.G., Wroblewski, D.A., and Tighe, S.N. Intelligent Assistance Through Collaborative Manipulation, *Proceedings of IJCAI 1991*, 9-14. (acceptance rate for 1991 unknown; typically in low 20s).
105. Terveen, L.G. and Wroblewski, D.A. A Collaborative Interface for Browsing and Editing Large Knowledge Bases, *Proceedings of AAAI 1990*, 491-496. 18% acceptance rate.

Book Chapters

106. Terveen, L., Riedl, J., Konstan, J., and Lampe, C. (2014) "Study, Build, Repeat: Using Online Communities as a Research Platform", in *Human Computer Interaction Ways of Knowing*, edited by Judith S. Olson and Wendy Kellogg, New York: Springer.
107. Amento, B., Harrison, C., Nathan, M., and Terveen, L. (2009), Asynchronous Communication – Fostering Social Interaction with CollaboraTV, in Cesar, P., Geerts, D., and Chorianopoulos, K. (ed.), *Social Interactive Television: Immersive Shared Experiences and Perspectives* (2009), Information Science Reference.
108. Amento, B., Terveen, L.G and Hill, W. From PHOAKS to TopicShop: Experiments in Social Data Mining, in Lueg, C. and Fisher, D. (ed.), *From Usenet to CoWebs: Interacting with Social Information Spaces* (2002), Springer.
109. Whittaker, S., Terveen, L.G, Hill, W.C., and Cherny, L. The Dynamics of Mass Interaction, in Lueg, C. and Fisher, D. (ed.), *From Usenet to CoWebs: Interacting with Social Information Spaces* (2002), (this chapter is a reprint of Whittaker et al 1998), Springer.
110. Terveen, L.G and Hill, W. Beyond Recommender Systems: Helping People Help Each Other, in Carroll, J. (ed.), *HCI in the New Millennium* (2001), Addison Wesley.
111. Terveen, L.G. Computer-Mediated Collaboration, in *More than Screen Deep: Toward Every-Citizen Interfaces to the Nation's Information Infrastructure* (1997), National Academy Press.

Other Publications

112. Masli, M. and Terveen, L. Geographical Social Production: Lessons from Cyclopath, in CHI 2013 GeoHCI Workshop.
113. Masli, M., Bouman, L., Owen, A., and Terveen, L. Geowiki + route analysis = improved transportation planning, CSCW 2013 Interactive Poster.
114. Brady, S. S., Sieving, R. E., Terveen, L. G., Rosser, B. R. S., Kodet, A. J., & Rothberg, V. D. (2012, October). *TeensTalkHealth: An interactive website to promote healthy relationships and prevent STIs*. Paper presented at the annual meeting of the American Public Health Association, San Francisco, CA.
115. Society for Personality and Social Psychology, San Antonio, TX. January, 2011. "The role of community orientation in promoting online participation" (with J. S. Kim, P. C. Dwyer, J. Filson Moses, P. T. Fuglestad, C. A. Mannino, R Davies, & M. Snyder)
116. Society for Personality and Social Psychology, San Antonio, TX. January, 2011. "Applying a functional approach to participation in online groups" (with P. T. Fuglestad, P. C. Dwyer, J. Filson Moses, J. S. Kim, C. A. Mannino, R Davies, & M. Snyder)
117. Society for Personality and Social Psychology, San Antonio, TX. January, 2011. "Past volunteerism predicts amount of content contributed in an online community" (with P. C. Dwyer, J. Filson Moses, P. T. Fuglestad, J. S. Kim, C. A. Mannino, R Davies, & M. Snyder)

118. Society for Personality and Social Psychology, San Antonio, TX. January, 2011. "Social motives and personality as predictors of online participation" (with J. Filson Moses, P. C. Dwyer, P. T. Fuglestad, J. S. Kim, C. A. Mannino, R Davies, & M. Snyder)
119. Kapoor, N., Frankowski, D., Konstan, J., and Terveen, L. Lessons Learned in Implementing the CHIplace Online Community, in *Proceedings of Human-Computer Interaction International 2005*.
120. Zhou, C., Ludford, P., Frankowski, D., and Terveen, L. An Experiment in Exploring How People Describe Places, Short Paper in *Proceedings of Pervasive 2005*.
121. Zhou, C., Ludford, P., Frankowski, D., and Terveen, L. An Experiment in Discovering Personally Meaningful Places from Location Data, Short Paper in *Proceedings of CHI 2005*.
122. Kapoor, N., Konstan, J., and Terveen, L. How Peer Photos Influence Member Participation in Online Communities, Short Paper in *Proceedings of CHI 2005*.
123. Amento, B., Hill, W., and Terveen, L. The Sound of One Hand: A Wrist-mounted Bio-acoustic Fingertip Gesture Interface, Short Paper in *Proceedings of CHI 2002*, 724-725. *33% acceptance rate*.
124. Terveen, L., Hill, W., and Amento, B. Collaborative Filtering to Locate, Comprehend, and Organize Collections of Websites, in *SIGART Bulletin*, 9, 3&4 (1998), 10-17.
125. Terveen, L., Hill, W., Amento, B., McDonald, D., and Creter, J. 1997. PHOAKS: a system for sharing recommendations. *Commun. ACM* 40, 3 (Mar. 1997), 59-62.
126. Terveen, L.G., Stolze, M., and Hill, W. From 'Model World' to 'Magic World', in *SIGCHI Bulletin*, 27, 4 (1995), 31-34.
127. Terveen, L.G., Papavero, E., and Tuomenoksa, M. DynaDesigner: A Tool for Rapid Design and Deployment of Device-Independent Interactive Services, Refereed Formal Demonstration in *Adjunct Proceedings of CHI'95*, 29-30.
128. Terveen, L.G. Person-Computer Cooperation through Collaborative Manipulation. Ph.D. Thesis, University of Texas Department of Computer Sciences, 1991.

Invited Presentations

Carnegie Mellon University

Northwestern University

The University of California Irvine

Carleton College

Twin Cities MetroGIS Policy Board

Hennepin County Bicycle Advisory Committee

University of Minnesota Digital Humanities Collaborative

University of Minnesota Advanced Transportation Technologies Seminar Series

Georgia Tech

University of Minnesota Urban Ecosystems Symposium

University of Minnesota New Media

IBM T.J. Watson Research

University of Illinois

University of Maryland

University of Washington
Carnegie-Mellon University
Swedish Institute of Computer Science / Royal Institute of Technology
Uppsala University, Sweden
New Jersey Institute of Technology
Microsoft Research
Vassar University
Rensselaer Polytechnic Institute
University of Colorado
Virginia Tech
University of Nebraska

Conference Presentations

- ACM Conference on Computer-Supported Cooperative Work (CSCW), 2014.
- ACM Conference on Computer-Supported Cooperative Work (CSCW), 2012.
- ACM Conference on Computer-Supported Cooperative Work (CSCW), 2002.
- ACM Conference on Computer-Supported Cooperative Work (CSCW), 1998.
- ACM Conference on Human Factors in Computing Systems (CHI), 1998.
- ACM Conference on Design of Interactive Systems, 1997.
- ACM Conference on Human Factors in Computing Systems (CHI), 1997.
- ACM Conference on Human Factors in Computing Systems (CHI), 1996.
- IFIP TC13 International Conference on Human-Computer Interaction (INTERACT), 1995.
- IEEE Conference on Knowledge-Based Software Engineering, 1994.
- ACM Conference on Human Factors in Computing Systems (CHI), 1993.
- Conference on Information and Knowledge Management, 1993.
- Conference on Information and Knowledge Management, 1992.
- National Conference on Artificial Intelligence (AAAI), 1991.
- International Joint Conference on Artificial Intelligence (IJCAI), 1991.
- National Conference on Artificial Intelligence (AAAI), 1990.

Workshop Presentations

- CHI 2013 GeoHCI Workshop
 - “Geographical Social Production: Lessons from Cyclopath”
- CSCW 2002 Workshop on “The Role of Place in Shaping Virtual Communities”:
 - “Place-Based Community Information Systems”.
- CHI 99 Workshop on “Interacting with Recommender Systems”:

- “Visualization Interfaces for Recommender Systems”.
- 1999 Human-Computer Interaction Consortium Workshop:
 - “A Reference Task Agenda for Human-Computer Interaction”.
- 1998 AAI Workshop on “Recommender Systems”:
 - “The PHOAKS Recommender System”.
- 1997 Human-Computer Interaction Consortium Workshop:
 - “The PHOAKS Recommender System”.
- 1995 Lifelike Computer Characters Workshop:
 - “Hidden Hands, not Talking Heads: The Magic World Interaction Paradigm”.
- 1995 Lifelike Computer Characters Workshop:
 - “Moving Agent-User Voice Dialogue towards Natural Conversation”.
- CHI 95 Workshop on “‘Model World’ to ‘Magic World’: Making Visual Objects the Medium for Intelligent Design Assistance”:
 - “The ‘Magic World’ Approach to Human-Computer Collaboration”.
- 1993 AAAI Fall Symposium: “Human-Computer Collaboration: Reconciling Theory, Synthesizing Practice”:
 - “A Framework for Human-Computer Collaboration”
- 1993 AI-ED Workshop on “Collaborative Problem Solving: Theoretical Frameworks and Innovative Systems”:
 - “Collaborative Problem Solving in Interactive Systems”.
- 1992 CAIA Workshop on “Applying AI To Software Problems: Assessing Promises and Pitfalls”:
 - “Representing and Disseminating Software Design Knowledge”.
- 1992 AAAI Spring Symposium on “Cognitive Aspects of Knowledge Acquisition”:
 - “In The Footprints of The Masters: Embedding Knowledge Acquisition in Organizational Activity”.
- AAAI 90 Workshop on “Complex Systems, Ethnomethodology, and Interaction Analysis”:
 - “Resources for Person-Computer Collaboration”.
- 1990 AAAI Spring Symposium on “Knowledge-Based Human-Computer Communication”:
 - “Tools for Human-Computer Collaboration”.

Patents

U.S. Patent # 5,388,188. Apparatus and methods for providing design advice. (with P. Selfridge). Issued February 7, 1995.

U.S. Patent #5,659,724. Interactive data analysis apparatus employing a knowledge base. (with A. Borgida, R.J. Brachman, T. Kirk, and P. Selfridge). Issued August 19, 1997.

U.S. Patent #5,680,530. Graphical environment for interactively specifying a target system (with P. Selfridge). Issued October 21, 1997.

U.S. Patent #5,806,060. Interactive data analysis employing a knowledge base. (with A. Borgida, R.J. Brachman, T. Kirk, and P. Selfridge). Issued September 8, 1998.

U.S. Patent # 5,809,492. Apparatus and method for defining rules for personal agents. (with L. Murray). Issued September 15, 1998.

U.S. Patent # 5,953,393. Personal Telephone Agent. (With P. Culbreth, P. Danielsen, R.J. Hall, E. Papavero, and M. Tuomenoksa). Issued September 14, 1999.

U.S. Patent #6,029,192. System and method for locating resources on a network using resource evaluations derived from electronic messages. (with W.C. Hill). Issued February 22, 2000.

U.S. Patent #6,244,873. Wireless myoelectric control apparatus and methods. (with W.C. Hill, F.C. Pereira, and Y. Singer). Issued June 12, 2001.

U.S. Patent #6,256,648. System and method for selecting and displaying hyperlinked information resources. (with W.C. Hill). Issued July 3, 2001.

CONSULTING

- Sound View Innovations, LLC v. LinkedIn Corp.
 - Retained as expert on behalf of LinkedIn by Klarquist Sparkman (2016)
- TiVo Inc. v. Samsung Electronics
 - Retained as expert on behalf of TiVo by Irell & Manella (2016)
 - Case No. 2:15-cv-1503 (E.D. Tex.)
- Netflix v. Rovi 4:11-cv-06591
 - Retained as expert on behalf of Netflix by Keker Van Nest (2014-2015)
 - Presented technology tutorial to Judge, March 2015
- Clear with Computers vs. Vermeer
 - *Clear with Computers, LLC vs. Vermeer Corporation*
Civil Action No. 13-cv-00167 (E.D. Texas)
 - Retained by Faegre, Baker, and Daniels
- Clear with Computer vs. Manitowoc Cranes, LLC.
 - *Clear with Computers, LLC vs. Vermeer Corporation*
Civil Action No. 13-cv-00167 and related cases (E.D. Texas)
 - Retained by Baker Botts
- Clear with Computers vs. Terex Corporation
 - *Clear with Computers, LLC v. Terex Corporation*
Civil Action No. 6:12-cv-00634 (E.D. Texas)
 - Retained by Fish & Richardson
- Clear with Computers vs. Valmont Industries, Inc.
 - *Clear with Computers, LLC v. Valmont Industries, Inc.*
Civil Action No. 6:13-cv-00166 (E.D. Texas)
 - Retained by Fish & Richardson
 - During the first half of 2014 I was retained by three different firms on behalf of four different clients to work on invalidity. I also worked on non-infringement on behalf of Vermeer.
- Intellectual Ventures vs. U.S. Bancorp and U.S. Bank
 - *Intellectual Ventures II LLC v. U.S. Bancorp and U.S. Bank*

- Civil Action No. 13-2071 (U.S. District of Minnesota, Fourth Division)
 - Retained by Winthrop & Weinstine on behalf of U.S. Bancorp and U.S. Bank
 - Consulted on non-infringement and invalidity from late 2013 through mid 2014
- “Products Containing Interactive Program Guide and Parental Control Technology”: Rovi vs. LG Electronics, Mitsubishi, Netflix, Roku, and Vizio.
 - Case No. ITC 337-TA-845
 - International Trade Commission
 - Retained as expert on behalf of respondents Netflix and Roku by Keker Van Nest (2012-2013)
 - Deposed: 2013
 - Testified in court: 2013
- Motorola Mobility vs. Microsoft and Microsoft vs. Motorola Mobility (Counterclaim)
 - Case No. 1:10-CV-24063-MORENO
 - United States District Court Southern District of Florida
 - Retained as expert on behalf of Microsoft (2011-2012) by Sidley Austin
 - Deposed: 2011
- Interval Licensing LLC vs. Aol, Inc.; Apple, Inc., eBay, Inc.; Facebook, Inc.; Google, Inc.; Netflix, Inc.; Office Depot Inc.; OfficeMax Inc.; Staples, Inc.; Yahoo, Inc.; and YouTube, LLC.
 - Case No. 2:10-cv-01385-MJP
 - United States District Court Western District of Washington at Seattle
 - Retained as expert of behalf of four of the respondents – eBay Inc.; Netflix, Inc.; Office Depot, Inc.; and Staples, Inc. – by Klarquist Sparkman (2011)
- Microsoft vs. Tivo
 - Case No. 5:10-cv-00240-RS
 - United States District Court Northern District of California
 - Retained as expert on behalf of plaintiff by Perkins Coie (2010-2011)
- Elsevier B.V., Elsevier Inc., and Mosby, Inc. vs. UnitedHealth Group, Inc. etc.
 - 09 Civ. 2124
 - United States District Court Southern District of New York
 - Retained as expert on behalf of respondent by Dorsey & Whitney (2010)
- NORTHBROOK DIGITAL LLC vs. Vendio Services, Inc.
 - Civil File No. 07-CV-2250
 - United States District Court District of Minnesota
 - Retained as expert on behalf of plaintiff by Dorsey & Whitney (2009)

SERVICE

External Professional Activities

ACM Special Interest Group on Computer Human Interaction

- President: 2015-2018
- Executive Committee: Adjunct Chair for Awards, 2012 – 2015.
- Executive Committee: Vice President for Membership and Communication, 2009 – 2012.

Conference Chair

- CHI 2002: ACM Conference on Human Factors in Computing Systems
- IUI 1998: ACM Conference on Intelligent User Interfaces.

Program Committee Chair

- CSCW 2004: ACM Conference on Computer-Supported Cooperative Work.
- CSCW 2013: ACM Conference on Computer-Supported Cooperative Work.

Computer Supported Cooperative Work (CSCW) Steering Committee

- Chair, 2012-2014 (First elected Chair of the committee that oversees the CSCW Conference and the general CSCW and Social Computing research community.)

Awards Committee Chair

- CSCW 2008: ACM Conference on Computer-Supported Cooperative Work.

Journals Edited

- Special Issue of *Knowledge-Based Systems* on Human-Computer Collaboration, Vol. 8, No. 2-3, 1995.

Proceedings Edited

- Proceedings of the 2002 ACM Conference on Human Factors in Computing Systems (CHI 2002).
- Proceedings of the 1998 ACM Conference on Intelligent User Interfaces (IUI 1998).

Editorial Boards

- Communications of the ACM, 2009-present.
- ACM Transactions on CHI, 2000-2006.
- Knowledge-Based Systems, 1993-present.
- ACM *intelligence*, 1998-2001.

Program Committees

- ACM Conference on Human Factors in Computing Systems (CHI): 1999-2004, 2006, 2016.
- ACM Conference on Computer-Supported Cooperative Work (CSCW): 2000, 2006, 2008, 2010, 2011, 2012, 2015.
- ACM Conference on Intelligent User Interfaces (IUI): 1997-2000, 2004.
- ACM Recommender Systems Conference: 2007, 2008, 2009, 2010, 2013.
- ACM GROUP Conference: 2007, 2009, 2010.
- ACM SIGIR Conference: 2008.
- AAAI Conference on the Web and Social Media: 2011, 2012, 2016.
- User Modeling, Adaptation, and Personalization: 2011, 2016.
- NordiCHI 2008, 2010.
- Communities and Technology: 2009.
- Computer-Supported Cooperative Learning 2002.
- User Modeling: 2001.
- International Conference on Knowledge Capture 2001.
- Knowledge-Based Software Engineering (KBSE): 1994-1999.
- Intelligent Data Analysis: 1997.
- National Conference on Artificial Intelligence (AAAI): 1996-1997.
- IEEE Conference on Artificial Intelligence for Applications (CAIA): 1994.
- International Workshop on Privacy-Aware Location-based Mobile Services: 2007.
- SIGIR Workshop on Future Challenges in Expertise Retrieval: 2008.

Other conference leadership positions

- ACM Conference on Computer-Supported Cooperative Work Doctoral Consortium Co-Chair, 2016 and 2011.
- The 2nd AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2014) Doctoral Consortium Co-Chair.
- International Symposium on Wikis and Open Collaboration Doctoral Symposium Chair, 2011.
- ACM Conference on Recommender Systems Doctoral Consortium, Co-Chair, 2007.
- International Joint Conference for Artificial Intelligence Doctoral Consortium, Co-Chair, 1997.
- American Association for Artificial Intelligence Doctoral Consortium, Co-Chair, 1997.
- American Association for Artificial Intelligence Doctoral Consortium, Co-Chair, 1996.

Reviewer

Journals

- ACM Computing Surveys.
- IEEE Transactions on Data and Knowledge Engineering.
- IEEE Expert.
- Information Systems.
- International Journal of Human-Computer Studies.
- Journal of Computer-Supported Cooperative Work.

Conferences

- Ninth IFIP International Conference on Human-Computer Interaction (INTERACT): 2003.
- ACM Conference on Human Factors in Computing Systems (CHI): 1995-1998.
- ACM Conference on Computer-Supported Cooperative Work (CSCW):1998.
- ACM Symposium on User Interface Software and Technology (UIST): 1996.
- ISSM Conference on Information and Knowledge Management: 1993.

National Science Foundations Panels

Served on panels in 1996, 1998, 2001, 2003, 2005, 2010, 2012, 2013.

Workshops Organized

- ACM Wikis and Open Collaboration Doctoral Symposium, 2011.
- Social Computational Systems Community Workshop, 2011.
- ACM Computer Supported Cooperative Work Doctoral Consortium, 2011.
- International Conference on Ubiquitous Computing (UbiComp): “Multi-device Interfaces for Ubiquitous Peripheral Interaction”, 2003.
- ACM Conference on Human Factors in Computer Systems (CHI): “Interacting with Recommender Systems”, 1999.
- International Joint Conference for Artificial Intelligence Doctoral Consortium, 1997.
- American Association for Artificial Intelligence Doctoral Consortium, 1997.
- American Association for Artificial Intelligence Doctoral Consortium, 1996.
- ACM Conference on Human Factors in Computer Systems (CHI): “‘Model World’ to ‘Magic World’: Making Visual Objects the Medium for Intelligent Design Assistance”, 1995.
- ACM Conference on Human Factors in Computer Systems (CHI): “New Uses and Abuses of Interaction History”, 1994.
- AAAI Fall Symposium: “Human-Computer Collaboration: Reconciling Theory, Synthesizing Practice”, 1993.
- World Conference on Artificial Intelligence and Education: “Collaborative Problem Solving:

Theoretical Frameworks and Innovative Systems”, 1993.

- Conference on AI for Applications: “Applying AI To Software Problems: Assessing Promises and Pitfalls”, 1992.

Other Professional Service

- Member of SIGCHI Publications Board and Conference Management Committee, 2002-2004.
- ACM Special Interest Group on Artificial Intelligence Conference Chair (1995-1999); originated and co-organized Doctoral Consortia held in conjunction with AAAI and IJCAI.

Internal Service

- Chair, Department Head Search Committee, 2015.
- Chair, Strategic Planning Committee, 2013-2014.
- Chair, Social Computing Faculty Search Committee, 2013-2014.
- Curriculum Committee, 2012-2013.
- Chair, Strategic Planning Committee / Faculty Recruiting Committee, 2011-2012.
- Director of Graduate Studies: 2007-2010.
- Communications Committee: 2007
- Newsletter/Brochure Committee (chair): 2006-2007
- Research Opportunities Committee: 2005-2007
- Curriculum Committee: 2003-2005.
- Hosted Robert Kraut, Cray Colloquium speaker: October 2003.
- Participated in recruiting and admission activities.
- Information, Technology, and Everyday Life Initiative: Committee Member.

TEACHING AND ADVISING

Ph.D. Students Advised

- Dan Cosley (co-advised with John Riedl): completed PhD July 2006, currently an Associate Professor at Cornell University; on leave as Program Director at NSF.
- Pamela Ludford: completed PhD September 2007, currently an independent consultant.
- Reid Priedhorsky: completed PhD August 2010, currently a Postdoctoral Research Associate at Los Alamos National Laboratories.
- Mikhail Masli: completed PhD July 2013, currently employed at IBM.
- Aaron Halfaker (co-advised with John Riedl): completed PhD September 2013, currently employed at the Wikimedia Foundation.
- Katie Panciera: completed PhD August 2014; currently employed at Google.
- Fernando Torre: completed PhD September 2014; founder of a startup.

- Shuo (Steven) Chang: completed PhD August 2016; currently employed at Quora.
- Tien Nguyen (co-advised with Joe Konstan): completed PhD August 2016; currently employed at Pinterest.
- Morten Warncke-Wang (co-advised with Brent Hecht): completed PhD December 2016.
- Jacob Thebault-Spieker (entered program Fall 2011).
- Vikas Kumar (entered program Fall 2011)
- Hannah Miller (entered program Fall 2013)
- Andrew Hall (entered program Fall 2014)
- Bowen Yu (entered program Fall 2014; co-advised with Haiyi Zhu).

M.S. Students Advised

- Tyler Danielsen – received degree in 2016
- Jie Kang – received degree in 2016
- Zahra Eslami – received degree in 2015
- Yanjie Liu – received degree in 2013
- Renji Yu – received degree in 2012
- Carol Drysdale – received degree in 2011
- Jingwen Zhang – received degree in 2011
- Jisu Oh – received degree in 2010
- Sara Drenner – received degree in 2008
- Anna Rouben – received degree in 2006
- Arjun Sundararajan – received degree in 2006
- Pamela Ludford – received degree in 2005
- Rahul Akolkar – graduated May 2004

Undergraduate Honors / Senior Thesis Students Advised

- Harmanprett Kaur, 2016
- Arlo Siemsen, 2014
- Johnathan Frenz, 2013
- David Pitchford - 2012
- Michael Ludwig - 2010
- Jordan Focht – 2010
- Kurt Wilms - 2005
- John Murphy – 2004

Other Committees

- Catherine Grevet, PhD.: member of preliminary and final examination committees (Georgia Tech).
- Loxley Wang, PhD: member of preliminary and final examination committees.
- Abigail Bakke, PhD: member of preliminary and final examination committees
- Tahir Sousa, MS: member of final examination committee
- Tony Lam, PhD: member of preliminary examination committee.
- Michael Janseen, PhD: member of preliminary examination committee.
- Haleh Hagh Shenaa, PhD: member of preliminary & final examination committees.
- Julie Beilfuss, MS: member of final examination committee.
- Liv Knatterud, MS: member of final examination committee.
- Sean McNee, MS: member of final examination committee.
- Shankar Subrahmanian, MS: member of final examination committee.
- Vamsee Venuturumilli, MS: member of final examination committee.
- Eric Gilbert, Ph.D. Preliminary and Final Examination Committees (UIUC, 2009/2010): member of thesis committee.
- Yi (Jenny) Zhang, Ph.D. (New Jersey Institute of Technology, 2004): member of thesis committee.
- Brian Amento, Ph.D. (Virginia Tech, 2001): member of thesis committee.
- David McDonald, Ph.D. (UC Irvine, 2000): member of thesis committee.

Courses taught

Semester	Course
Fall 2016	CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2016	CSCI 5125: Collaborative and Social Computing
Fall 2015	CSCI 1133H: Introduction to Computer Science (Honors) CSCI 8115: Human-Computer Interaction and UI Technology
Spring 2015	CSCI 5125: Collaborative and Social Computing
Fall 2014	CSCI 5115: User Interface Design, Implementation, and Evaluation HSEM 2519H: Honors Seminar on Crowdsourcing
Spring 2014	CSCI 8115: Human-Computer Interaction and UI Technology
Fall 2013	CSCI 1901H: Introduction to Computer Science (Honors) CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2013	CSCI 5125: Collaborative and Social Computing SEng 5115: User Interface Design and Evaluation
Fall 2012	CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2012	CSCI 8115: Human-Computer Interaction and UI Technology SEng 5115: User Interface Design and Evaluation
Fall 2011	CSCI 1902: Structure of Computer Programming II CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2011	CSCI 5125: Collaborative and Social Computing SEng 5115: User Interface Design and Evaluation
Fall 2010	CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2010	CSCI 8115: Human-Computer Interaction and UI Technology
Fall 2009	CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2009	CSCI 5125: Collaborative and Social Computing SEng 5115: User Interface Design and Evaluation
Fall 2008	CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2008	CSCI 1902: Structure of Computer Programming II SEng 5115: User Interface Design and Evaluation
Fall 2007	CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2007	SEng 5115: User Interface Design and Evaluation
Spring 2007	CSCI 5125: Collaborative and Social Computing
Fall 2006	CSCI 1902: Structure of Computer Programming II CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2006	CSCI 8115: Human-Computer Interaction and UI Technology
Fall 2005	CSCI 1902: Structure of Computer Programming II
Spring 2005	CSCI 5116: GUI Toolkits and Their Implementation

Fall 2004 CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2004 CSCI 5980: Collaborative Computing
Fall 2003 CSCI 5115: User Interface Design, Implementation, and Evaluation
Spring 2003 CSCI 5116: GUI Toolkits and Their Implementation
Fall 2002 CS 8115: Human-Computer Interaction and UI Technology

Tutorials

“Intelligent User Interfaces: Issues, Approaches, Evaluation”, offered at 1993 Conference on Artificial Intelligence for Applications.