

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC., AMAZON.COM SERVICES LLC,

Petitioner,

v.

NOKIA TECHNOLOGIES OY,

Patent Owner.

Case No. IPR2024-01043

U.S. Patent 8,918,741

Declaration of Benjamin B. Bederson, Ph.D.

TABLE OF CONTENTS

I.	THE '741 PATENT	7
A.	Overview	7
B.	Prosecution History	8
C.	District Court Litigation	9
D.	Priority Date	9
E.	Challenged Claims	10
II.	LEVEL OF ORDINARY SKILL IN THE ART	10
III.	CLAIM CONSTRUCTION	12
IV.	INVALIDITY GROUNDS.....	13
A.	Grounds 1-2	13
1.	U.S. Patent Application Publication No. 2007/0180492 ("Hassan") (Ex-1004)	14
2.	U.S. Patent Application Publication No. 2006/0246955 ("Nirhamo") (Ex-1005).....	16
3.	Independent Claims 1, 11, and 17.....	18
4.	Dependent Claims 2 and 13	39
5.	Dependent Claims 3 and 14	42
6.	Dependent Claim 4.....	45
7.	Dependent Claims 5 and 15	48
8.	Dependent Claim 6.....	49
9.	Dependent Claim 7.....	51
10.	Dependent Claim 8.....	52
11.	Dependent Claims 9 and 16	54

12.	Dependent Claim 10	56
13.	Dependent Claims 12 and 18	57
14.	Dependent Claim 19	58
15.	Dependent Claim 20	58
B.	Grounds 3-4	58
1.	U.S. Patent No. 7,689,939 (“Becker”) (Ex-1006)	59
2.	Independent Claims 1, 11, and 17	63
3.	Dependent Claims 2 and 13	81
4.	Dependent Claims 3 and 14	84
5.	Dependent Claim 4	87
6.	Dependent Claims 5 and 15	90
7.	Dependent Claim 6	91
8.	Dependent Claim 7	92
9.	Dependent Claim 8	93
10.	Dependent Claims 9 and 16	95
11.	Dependent Claim 10	97
12.	Dependent Claims 12 and 18	98
13.	Dependent Claim 19	98
14.	Dependent Claim 20	100
V.	BACKGROUND AND QUALIFICATIONS	100
A.	Compensation	113
B.	Materials and Other Information Considered	114
VI.	UNDERSTANDING OF THE LAW	114

VII. RESERVATION OF RIGHTS.....	114
---------------------------------	-----

I, Benjamin Bederson declare as follows:

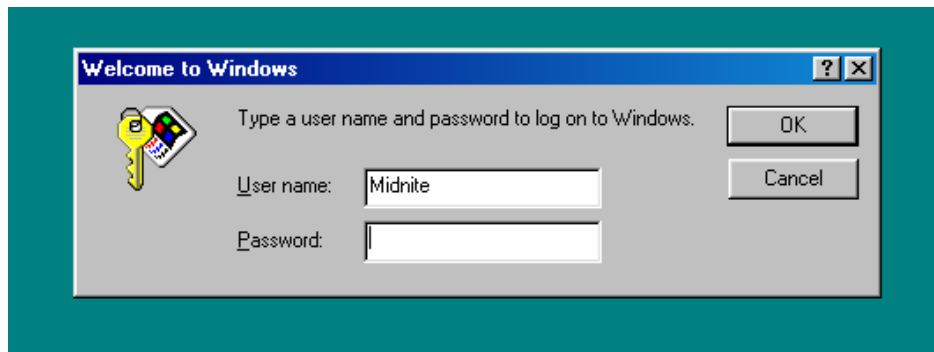
1. My name is Benjamin Bederson. I have been a Professor of Computer Science at the University of Maryland (“UMD”), where I have had joint appointments at the Institute for Advanced Computer Studies and the College of Information Studies (Maryland’s “iSchool”) since 1998 and am currently Professor Emeritus. I have prepared this report as an expert witness retained by Amazon.com, Inc and Amazon.com Services LLC. In this report I give my opinions as to whether certain claims of U.S. Patent No. 8,918,741 (“the ’741 patent”) are invalid. I provide technical bases for these opinions as appropriate.

2. This report contains statements of my opinions formed to date and the bases and reasons for those opinions. I may offer additional opinions based on further review of materials in this case, including opinions and/or testimony of other expert witnesses. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters contained herein.

OVERVIEW OF THE TECHNOLOGY

3. For many decades, computers have had locked and unlocked modes. Many computers, going back to Unix and early Windows computers of the 1990s, required a password to unlock the computer, to prevent unauthorized access when

the user was away. Below is an exemplary Windows 98 prompt for unlocking a PC:¹



4. By the 2000s, mobile computers, including mobile phones, also included locked and unlocked modes to enhance user security and convenience. In the locked mode, the mobile devices were protected by passwords, PINs, or biometric authentication, ensuring that only authorized users could access the device's contents. Conversely, the unlocked mode allowed users full access to the device's functionality, enabling seamless communication, internet browsing, and application use. This dual-mode operation balanced security needs with the growing demand for mobile productivity and connectivity.

5. Several scenarios were recognized in the art where users wanted limited functionality without having to unlock their computers/phones. For example, a device owner may want to share his/her device with co-workers for the purpose of allowing them to view a particular work related message or email

¹ I found this image at <https://guidebookgallery.org/screenshots/win98>. It is consistent with my personal memory of the Windows 98 login screen.

without allowing access to other applications. *E.g.*, Ex-1004, ¶¶1-3. As another example, a device owner may only wish to share the device with children for the purpose of gaming without allowing access to other applications. *Id.* Therefore, the prior art included numerous examples of PCs and other devices that allowed limited interaction with certain applications without requiring that the device be unlocked. This included limited access to messaging, email, and PC calling applications while the device remained locked. *See, e.g.*, Ex-1004, ¶¶13-14, ¶40; Ex-1006, 2:61-3:11.

I. THE '741 PATENT

A. Overview

6. The '741 patent “relate[s] to user interfaces and in particular to transitioning from an idle/locked state of a device to an active/unlocked state of a device.” Ex-1001, 1:16-18. However, the patent does not purport to invent locked mode for devices. It admits that locked mode was known in the art, including, for example, a screen saver. Ex-1001, 1:23-25 (“Background” section), 4:17-19. Nor does the '741 patent purport to invent the conventional process of unlocking locked devices, admitting that it was known in the art. Ex-1001, 1:28-30 (“Background” section).

7. Instead, the '741 patent purportedly allows users to interact with applications, without fully activating them, while the device remains locked. For

example, the patent discusses user operations for “unlocking a touch screen device” using “a movable object ... to select and highlight different regions or areas on the screens” and “determine a state, utility or function” of applications “without the need to unlock the device or open the application.” *See, e.g.*, Ex-1001, Abstract, 4:7-8, 6:57-59, Fig. 2A.

8. The specification includes an example where the user may view the current time by selecting the icon of the clock application. Ex-1001, 6:56-7:24.

9. However, as explained above, it was well known in the art to select and interact with interface objects while a device remains locked. This was known for mobile phones (explained for Ground 1 below) and PCs (explained for Ground 2). The '741 patent also references other computer features that were common and conventional years before the patent was filed, including, e.g., touch screens, pointing devices, and dimming. *See, e.g.*, Ex-1001, 4:5-30, 3:18-21; Ex-1007.

B. Prosecution History

10. The '741 patent was filed as Application No. 13/889,750 and allowed after two Office Actions. In the first Office Action, the Examiner rejected the claims as being anticipated and obvious over the cited prior art. Ex-1002, 000069-75. In addition, the Examiner rejected the original claim 5 under Section 112, finding that meaning of the term “further icon” is unclear. In response, the Applicant argued that at least two of the cited references do not properly qualify as

prior art, and clarified the meaning of “further icon.” In the second Office Action, the Examiner withdrew the first rejection under Sections 102 and 103 but maintained the rejection under Section 112. The Examiner rejected the claims as being anticipated and obvious over some other prior art. Ex-1002, 000708-715. In response, the Applicant cancelled claim 5, and amended the independent claims, replacing “icon” with “object” and adding “without fully activating the application” into the claims. Ex-1002, 000745-749. This led to the allowance of the ’741 patent. Ex-1002, 000835.

C. District Court Litigation

11. In district court, Patent Owner (“PO”) accused tablet products with broad allegations that the ’741 patent is satisfied by user interface elements that users can interact with when the device is locked, including for example, elements related to Wi-Fi controls, Bluetooth controls, screen brightness control, and calendar application. *See* Ex-1008, 000113, 000116-117; Ex-1009, 000002-10.

D. Priority Date

12. The ’741 patent was filed May 8, 2013. The ’741 patent is a continuation that cites an original application from January 24, 2012, and a provisional application dated June 29, 2007. For purposes of this Declaration, I have analyzed obviousness as of June 29, 2007. I do not offer an opinion as to whether the ’741 patent is entitled to a certain priority date. My invalidity opinions

would not change if a later date (e.g., January 24, 2012) was determined to be the correct priority date because the prior art relied upon in this declaration would still be prior art.

E. Challenged Claims

13. I understand that Petitioner is challenging the validity of claims 1-20 of the '741 patent in the Petition for Inter Partes Review to which this declaration will be attached. Those claims are reproduced in Appendix 3. While this Petition and declaration are directed to the challenged claims, I have considered all claims 1-20 of the '741 patent, as well as portions of the '741 patent prosecution history in forming my opinions.

II. LEVEL OF ORDINARY SKILL IN THE ART

14. I have analyzed the '741 patent and determined that the field of the patent is user interfaces. *See, e.g.*, Ex-1001, 1:16-18. The specification describes the background of the invention primarily in terms of user interface, including user interface operations such as unlocking devices. Ex-1001, 1:15-37. The summary of the invention discusses user interface operations such as contacting screens to reveal application icons associated with an active/unlocked state of the device. Ex-1001, 1:41-67. The claims are also directed to operations on user interfaces.

15. In determining the characteristics of a hypothetical person of ordinary skill in the art ("POSITA") of the '741 patent at the time of the claimed invention,

I considered several things, including various prior art techniques relating to user interfaces, the type of problems that such techniques gave rise to, and the rapidity with which innovations were made. I also considered the sophistication of the technologies involved, and the educational background and experience of those actively working in the field at the time. I also considered the level of education that would be necessary to understand the '741 patent. Finally, I placed myself back in the relevant period of time and considered the engineers and programmers that I have worked with and managed in the field of user interfaces.

16. I came to the conclusion that a person of ordinary skill in the field of art of the '741 patent would have been a person with a bachelor's degree in electrical engineering, computer engineering, computer science, or a similar field with two to three years of experience in user interfaces or a person with a master's degree in electrical engineering, computer engineering, computer science, or a similar field with a specialization in user interfaces. A person with less education but more relevant practical experience may also meet this standard.

17. In the relevant timeframe (2006-2007), a POSITA would have known and had the skills necessary to program and build systems that meet all limitations of the challenged claims, including the following:

- Using and applying known user interface operations and techniques, including, for example, locking/unlocking devices, adjusting

luminance of elements (e.g., by highlighting) displayed on user interfaces, touch screens, and other known operations and techniques;

- Designing systems that allow various types of user interactions in graphical user interfaces, including, for example, selecting objects displayed on the user interface to initiate or activate associated application or functionalities.

III. CLAIM CONSTRUCTION

18. For purposes of this inter partes review, I have considered the claim language, specification, and portions of the prosecution history, to determine the meaning of the claim language as it would have been understood by a person of ordinary skill in the art at the time of the invention. The “plain and ordinary meaning” or *Phillips* standard has traditionally been applied in district court litigation, where a claim term is given its plain and ordinary meaning in view of the specification from the viewpoint of a person of ordinary skill in the art at the time of the patent.

19. I have applied the *Phillips* standard in my analysis. Unless otherwise stated, I have applied the plain and ordinary meaning to claim terms.

IV. INVALIDITY GROUNDS

20. There are a number of patents and publications that constitute prior art to the '741 patent. I have reviewed and considered the prior art discussed in this section, along with the materials listed in Appendix 2.

21. Based on my review and analysis of the materials cited herein, my opinions regarding the understanding of a POSITA in the relevant timeframe, and my training and experience, it is my opinion that the challenged claims of the '741 patent are invalid in view of the following grounds:

Grounds	Challenged Claims	Statutory Basis	Prior Art
1	1, 5-7, 10, 11, 15, 17, 19	§ 103	Hassan
2	1-20	§ 103	Hassan, Nirhamo
3	1, 5-7, 10, 11, 15, 17, 19	§ 103	Becker
4	1-20	§ 103	Becker, Nirhamo

A. Grounds 1-2

22. It is my opinion that a POSITA would have found the Challenged Claims obvious based on Hassan (Ground 1). Several dependent claims recite simple limitations including, for example, highlighting or illuminating an object (claims 2, 4), using a movable object (claim 3), using a touch screen display (claim 9), and displaying a dimmed icon (claim 12). Those claims are taught by the

combination of Hassan and Nirhamo (Ground 2), which together render claims 1-20 obvious.

23. Hassan teaches a mobile device that allows limited access to a communication application while the mobile device is in a locked state. *Infra* §IV.A.3. A POSITA would have been motivated to combine Hassan with Nirhamo's teachings for, e.g., dimming items in the display and including a touch screen in a mobile device. *Infra* §§IV.A.6 (explaining application of Nirhamo's dimming teachings to Hassan), IV.A.11 (explaining application of Nirhamo's touch screen teachings to Hassan). These had been standard features in computers for many years before 2007. Therefore, the combination of Ground 2 combines known, prior art elements according to known methods to yield predictable results (e.g., a touch screen that dims objects on the display during a locked state).

24. The motivation and reasonable expectation of success in combining these prior art elements is explained in more detail below. *See infra* §§IV.A.4, IV.A.6, IV.A.11.

**1. U.S. Patent Application Publication No. 2007/0180492
("Hassan") (Ex-1004)**

25. I have reviewed the Hassan reference. I understand that Hassan was not cited or considered during prosecution of the '741 Patent, based primarily on the fact that Hassan is not cited on the face of the patent, nor have I seen the reference discussed in the prosecution history.

26. Hassan was published August 2, 2007, and was filed as application 11/344,072 on February 1, 2006 and issued as a patent. I understand Hassan is prior art under at least pre-AIA §102(e) because it issued as a patent and was filed before the earliest possible filing date for the '741 patent, its provisional application's filing date of June 29, 2007.

27. Hassan “relates to electronic user interactive devices and, in particular, to methods and systems for enabling sharing of such devices with other users without compromising security.” Ex-1004, ¶1. Hassan discloses “[a] device and method for placing the device in a locked state having an associated set of permitted tasks so as to permit the device owner to share the device with others but maintain security over aspects of the device[.]” Ex-1004, Abstract, Fig. 3 (showing one embodiment of Hassan):

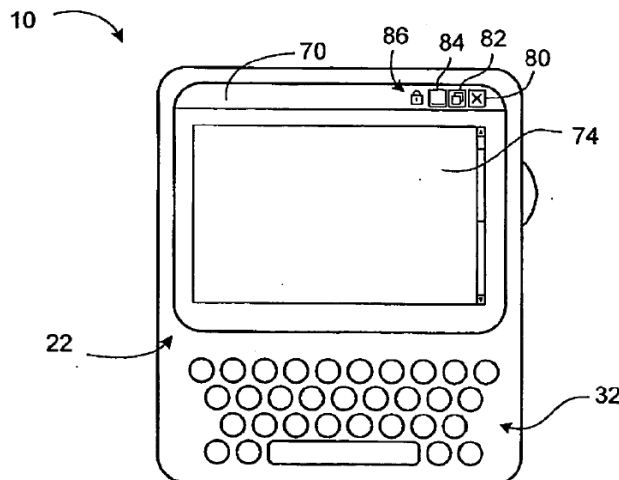


FIG. 3

28. Hassan confirms that causing a display of a device to display an object relating to an application was conventional and known in the prior art. Hassan teaches displaying icons related to applications when the device (e.g., mobile device) is in locked state. For example, Hassan teaches icons for launching an application and displaying the icon when the device is in a locked state. *E.g.*, Ex-1004, ¶59. In addition, Hassan teaches displaying an icon representing operations within an application for users to request or initiate those operations when the mobile device is in a locked state. Ex-1004, ¶¶70-72, ¶36. Hassan allows users to select these icons when the device is in a locked state, which causes the device to “evaluate the lock state of the device, the requested operation, and the permitted or impermissible tasks to determine whether to authorize the operation or instruct the lock module 260 to obtain authorization input.” Ex-1004, ¶¶70-72.

29. Hassan is in the same field of endeavor as the ’741 patent because it is directed to user interfaces. *See supra* §II; *e.g.*, Ex-1004, ¶1 (“The present application relates to electronic user interactive devices and, in particular, to methods and systems for enabling sharing of such devices with other users without compromising security.”).

**2. U.S. Patent Application Publication No. 2006/0246955
 (“Nirhamo”) (Ex-1005)**

30. I have reviewed the Nirhamo reference. I understand that Nirhamo was not cited or considered during prosecution of the ’741 Patent, based primarily

on the fact that Nirhamo is not cited on the face of the patent, nor have I seen the reference discussed in the prosecution history.

31. Nirhamo was published November 2, 2006, and was filed as application 11/120,319 on May 2, 2005. I understand Nirhamo is prior art under at least pre-AIA §102(a) because it was published before the earliest possible filing date for the '741 patent, its provisional application's filing date of June 29, 2007.

32. Nirhamo "relates to improved presentation, navigation, selection and/or operation options for portable communication devices, particularly, user interface options involved on the display screens thereof." Ex-1005, ¶1. Nirhamo discloses device operations, including highlighting items on the display when they are selected. Ex-1005, Abstract, ¶4, ¶¶29-31, ¶¶36-39, ¶41, ¶51, ¶71. For example, Nirhamo teaches highlighting a particular icon or element on the screen when it is selected by the user. Ex-1005, ¶29 ("Note, the example of FIG. 2A provides some additional information as for example when a particular icon or element 29 is selected. This element 29 may then be highlighted in some way (as here by a distinctively colorized box framed therearound, although other means are also usable herewith)."), ¶51, Fig. 2A.

33. Nirhamo also teaches dimming items that are unavailable or otherwise incompatible with an operation. Ex-1005, ¶36, ¶¶53-54 ("... Here only the currently focused item is de-emphasized or dimmed when the action indication is

focused on ‘Cancel.’ See ‘Item 2’ element number 82 b. This de-emphasis shows the direct relationship of the action ‘Cancel’ to the highlighted item 82 b, specifically, that the action is not applicable to the item.”), ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C.

34. Nirhamo is in the same field of endeavor as the ’741 patent because it is directed to user interfaces. *See supra* §II; Ex-1005, ¶1 (“This invention relates to improved presentation, navigation, selection and/or operation options for portable communication devices, particularly, user interface options involved on the display screens thereof.”).

3. Independent Claims 1, 11, and 17

35. Beyond their preambles, these claims recite identical limitations and are therefore addressed together.

[1a]. An apparatus, comprising: at least one processor; and at least one memory storing computer program code configured, working with the at least one processor, to cause at least the following to be performed:

36. Hassan teaches an apparatus (e.g., a mobile device). Ex-1004, ¶¶5-10, ¶20 (“[T]he present application may be embodied within a variety of user devices, including mobile devices, handheld devices, mobile telephones, personal digital assistants (PDAs) and other such devices.”), ¶43, ¶52, ¶68, Fig. 3 (showing a front view of the display of one embodiment of a mobile device):

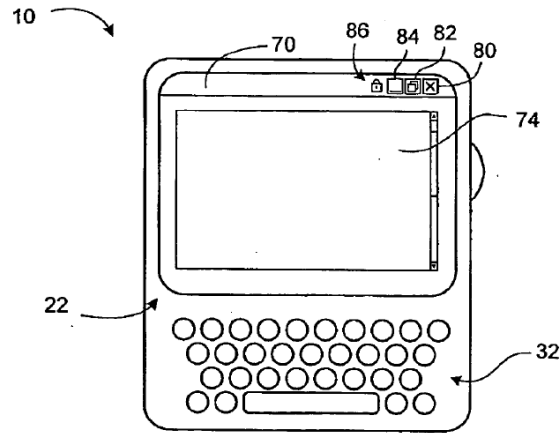


FIG. 3

37. Hassan implements its teachings with a mobile device that comprises a processor and memory that stores computer program code (e.g., software module). *E.g.*, Ex-1004, ¶21 (“Portions of the method and device described herein may be implemented in software. In this regard, the description below makes reference to many software elements that may be labeled as ‘modules’, ‘components’, etc. It will be appreciated that some of the described modules, components, programming objects, etc., or other software elements herein may in some cases be implemented together as a single module, component, application, subroutine, object, etc., or separately, and the precise implementation is a matter of design choice.”), ¶15 (“The device includes a main body having a user input device, a display, memory, and a processor operatively coupled to the user input device, the display and the memory.”), ¶17 (“The device includes a main body housing a user input device, a display, memory, and a processor operatively

coupled to the user input device, the display and the memory. The device also includes an operating system running on the processor for managing input and output through the user input device and the display and providing a platform for operation of application programs, wherein the operating system is configured to receive a user request to change to a selected task.”), ¶26 (“The device 10 includes a microprocessor 38 that controls the overall operation of the device. The microprocessor 38 interacts with the communications subsystem 11 and also interacts with further device subsystems such as a graphics subsystem 44, flash memory 24, random access memory (RAM) 26, auxiliary input/output (I/O) subsystems 28, serial port 30, keyboard or keypad 32, speaker 34, microphone 36, a short-range communications subsystem 40, and any other device subsystems generally designated as 42. The graphics subsystem 44 interacts with the display 22 and renders graphics or text upon the display 22.”), ¶27 (“Operating system software 52 and various software applications 54 used by the microprocessor 38 are, in one example embodiment, stored in a persistent store such as flash memory 24 or similar storage element. Those skilled in the art will appreciate that the operating system 52, software applications 54, or parts thereof, may be temporarily loaded into a volatile store such as RAM 26. It is contemplated that received communication signals may also be stored to RAM 26.”), ¶¶28-29 (“The

microprocessor 38, in addition to its operating system functions, preferably enables execution of software applications 54 on the device.”), Fig. 1:

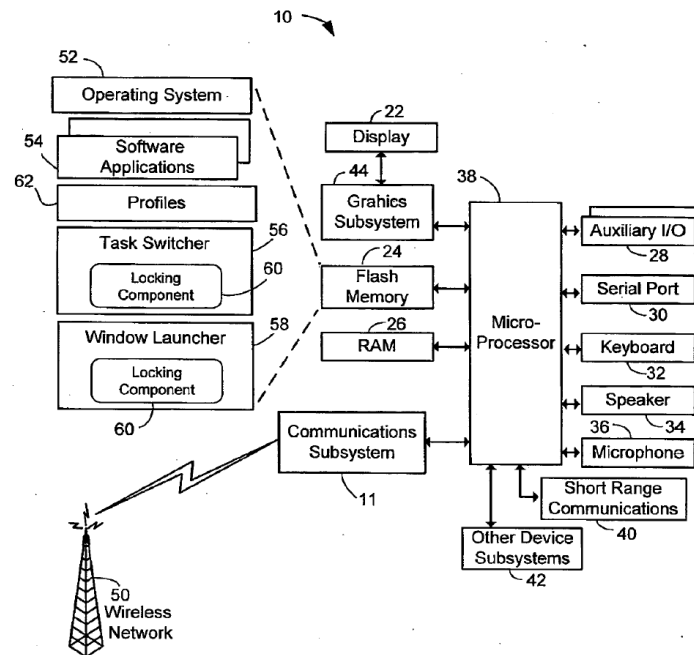


FIG. 1

Therefore, Hassan teaches its mobile device as an apparatus that comprises such a processor and memory.

38. Hassan’s computer program code is configured, working with the processor, to cause its teachings to be performed, including for the below-described steps for limitations [1b]-[1c] and claims 2-10. Ex-1004, ¶21, ¶15, ¶17, ¶¶26-29, Fig. 1. Additionally, a POSITA would have been knowledgeable about basic computer operation and understood, based on Hassan’s teachings, that its processor executes computer program code in memory to cause the device to carry

out its programmed functionality. That is how computers have operated for decades.

39. For the reasons set forth above, Grounds 1-2 teach the preamble and additional limitations of element [1a].

[11a]. A method, comprising:

40. Hassan teaches a method, including operations as described for limitations [11b]-[11c] and claims 12-16. *E.g.*, Ex-1004, Abstract (“A device and *method* for placing the device in a locked state having an associated set of permitted tasks so as to permit the device owner to share the device with others but maintain security over aspects of the device.”), ¶1 (“The present application relates to electronic user interactive devices and, in particular, to *methods* and systems for enabling sharing of such devices with other users without compromising security.”), ¶¶8-10, ¶13 (“The present application provides a device and a *method* for placing a device in a locked S=state having an associated set of permitted tasks so as to permit the device owner to share the device with others but maintain security over aspects of the device.”), ¶14, ¶16, ¶53, ¶59, ¶64. Limitations [11b]-[11c] are identical to limitations [1b]-[1c]. As explained below, Hassan teaches operations described for limitations [11b]-[11c]. *Infra* §§IV.A.3[11b]-[11c].

41. For the reasons set forth above, Grounds 1-2 teach this preamble.

[17a]. A non-transitory computer readable medium storing computer program code that, when performed by at least one processor, causes at least the following to be performed:

42. Hassan teaches implementing its methods using program code (e.g., software module) residing on computer readable storage medium (e.g., main memory of a mobile device) that is performed by a processor. Ex-1004, ¶21, ¶15 ¶17, ¶¶26-29 (“The microprocessor 38, in addition to its operating system functions, preferably enables execution of software applications 54 on the device. ... Further software applications 54 may also be loaded onto the device 10 through the network 50, an auxiliary I/O subsystem 28, serial port 30, short-range communications subsystem 40 or any other suitable subsystem 42, and *installed by a user in the RAM 26 or a non-volatile store for execution by the microprocessor 38.*”), Fig. 1. In addition, a POSITA would have understood that the types of storage medium disclosed by Hassan, including the main memory of a mobile device, are forms of non-transitory computer readable storage medium, as opposed to transitory signals. Therefore, Hassan teaches a non-transitory computer readable medium storing computer program code that, when performed by at least one processor, causes at least the operations as described for limitations [17b]-[17c] and claims 18-20 to be performed.

43. For the reasons set forth above, Grounds 1-2 teach this preamble.

[1b]/[11b]/[17b] causing a display of a device to display an object, relating to an application, when the device is in a locked mode; and

44. Hassan teaches limitations [1b], [11b], and [17b]. As explained above, Hassan teaches **a device** (e.g., a mobile device). *Supra* §IV.A.3[1a]; Ex-1004, ¶¶5-10, ¶20, ¶43, ¶52, ¶68, Fig. 3. Hassan teaches **causing a display of a device to display an object** (e.g., an icon), **relating to an application, when the device is in a locked mode**. Hassan teaches two types of icons that are related to an application: (1) icons for launching an application, and (2) icons for operations within an application. Both satisfy the claims.

45. First, Hassan teaches icons² for launching an application. Hassan causes the mobile device to display an icon for an application, which the user uses to request the launch of a new application, when the device is in a locked state. *E.g.*, Ex-1004, ¶59 (“Reference is now made to FIG. 5, which shows, in flowchart form, an embodiment of a method 120 for managing task change requests *in a locked device*. The method 120 begins with the receipt of a task change request in

² The '741 patent uses “object” broadly and confirms that icons fall within its scope. *E.g.*, '741 patent, 5:15-52 (“Although items 202 and 204 are described as icons, in alternate embodiments these could also be text or an object other than an icon.”), 6:30-55 (“The idle screen can present images, icons, text and objects, for example, that when selected and/or activated, provide direct access or short cuts to certain programs and applications associated with the idle state of the device.”), 7:64-8:18 (“Application and content objects can be provided by the menu control system 124.”).

step 122. The task change request may come from a variety of sources, as discussed above. For example, the user may attempt to launch *a new application from a start menu or icon.*)³ The icon represents and is related to the application. A POSITA would have understood this because the user launches the application from the icon, which is displayed on the device for the user to select. *Id.*

46. Second, Hassan teaches displaying an icon representing operations within an application for users to request or initiate those operations, including when the mobile device is in a locked state. For example, Hassan teaches an e-mail application that is associated with a set of tasks or operations such as forwarding, replying to, and opening messages, and “[t]he user may request or initiate the operation or task by selecting a menu item, clicking an icon associated with the operation, or otherwise providing user input to trigger the operation.” Ex-1004, ¶¶70-72, ¶36 (“In another aspect, references to a task changer or task change module are intended to encompass the object or software element at an application level or operating system level for initiating instructions to execute an operation within a specific application. For example, in the e-mail application, a user may request a certain function like forwarding a message, replying to a message, opening a message, through an appropriate user input (like *clicking on a designated icon*).”), Abstract (“The permitted tasks may designate specific

³ All annotations/emphasis added unless otherwise noted.

applications, specific operations or functions within applications or at the operating system level, one or more currently open windows, and other levels of granularity.”). Hassan allows the user to select these icons when the device is in a locked state, which as further explained below causes the device to “evaluate the lock state of the device, the requested operation, and the permitted or impermissible tasks to determine whether to authorize the operation or instruct the lock module 260 to obtain authorization input.” Ex-1004, ¶¶70-72. These icons are related to the application because they represent operations within the application and are displayed as part of the graphical user interface for the application. *See id.*

47. Third, Hassan teaches a **locked mode** of the device (e.g., a mobile device). For example, Hassan discloses “[a] device and method for placing the device in a *locked state* having an associated set of permitted tasks so as to permit the device owner to share the device with others but maintain security over aspects of the device[.]” Ex-1004, Abstract. Hassan thus teaches a locked mode that prevents normal access to the device and only allows user access to permitted tasks. Ex-1004, Abstract (“A device and method for placing the device in a locked state having an associated set of permitted tasks so as to permit the device owner to share the device with others but maintain security over aspects of the device.”), ¶13 (“The user of a locked device may only perform tasks that are permitted without unlocking the device.”), ¶14 (“The device has an operating system and the device

is placed in a locked state. The locked state is associated with permitted task information.”), ¶15 (“The device has a locked state and an unlocked state. The locked state is associated with permitted task information.”), ¶¶16-18, ¶¶38-39 (“In yet another embodiment, the locked state may be associated with a set of permitted tasks or a set of impermissible tasks, and the requested change may be evaluated to determine whether it is permitted.”). The locked state is associated with a set of permitted tasks that users other than the owner are allowed to access. Ex-1004, ¶13. As explained below, Hassan allows users to access permitted tasks within an application when the device (e.g., a mobile device) is in the locked state. *Infra* §IV.A.3[1c]/[11c]/[17c]; Ex-1004, ¶42, ¶38.

48. Once locked, the device remains in the locked state where only limited tasks are permitted, until the user unlocks the device. Other tasks are not allowed in the locked state; if the user requests a non-permitted task, the device must be unlocked to perform that task, which requires verification of the user’s authorization data: “[a] task change request is evaluated to determine whether the requested task is permitted and, if not, then an authorization window is opened to prompt the user to input authorization data. Upon verification of the authorization data, the device may be unlocked and the requested change executed.” Ex-1004, ¶13, ¶50 (“The failure of the requested task change may prompt the locking component 60 to obtain user authorization to unlock the device 10 in order to allow

the requested task change to occur.”), ¶52 (“In yet another embodiment, a user may click upon the lock indicator 86 to trigger the locking component 60 to bring up a user authorization window for unlocking the device 10 or altering the profile data 62.”), ¶66, Fig. 6.

49. The ’741 patent confirms that Hassan’s locked state falls within its scope.⁴ For the reasons set forth above, Hassan teaches limitations [1b]/[11b]/[17b]. Grounds 1-2 therefore teach limitations [1b]/[11b]/[17b].

[1c]/[11c]/[17c] enabling a user to select the object while the device is in the locked mode, in order to determine a state, function or utility of the application without fully activating the application while the device remains in the locked mode.

50. Hassan teaches limitations [1c], [11c] and [17c]. Hassan enables **a user to select the object (e.g., icon) while the device (e.g., mobile device) is in the locked mode**. As explained above, Hassan teaches two types of icons: (1) for launching an application, and (2) for requesting or initiating operations within an

⁴ For example, like Hassan, the ’741 patent’s embodiments allow limited user interaction to permitted applications when in locked mode. *See* ’741 patent, 11:32-45 (“The disclosed embodiments generally provide for a user to select an application or set of user interface elements that a device will open to from an idle and locked mode.”), 1:41-53 (“In one embodiment the method includes providing a touch or proximity screen device in an idle mode. ... The term idle mode generally be referred to as a reduced active region that can be manipulated in limited ways, such as by the unlocking gesture or interaction described herein.”).

application. *Supra* §IV.A.3[1b]/[11b]/[17b]. Both satisfy limitations [1c], [11c] and [17c].

51. First, for icons that launch an application, Hassan enables a user to select the icon representing an application to launch or attempt to launch the application when the mobile device is in a locked mode. Hassan allows the user to request a task change, which includes “launch[ing] a new application from a start menu or icon[,]” when the device is locked. Ex-1004, ¶59 (“Reference is now made to FIG. 5, which shows, in flowchart form, an embodiment of a method 120 for managing task change requests in a locked device. The method 120 begins with the receipt of a task change request in step 122. The task change request may come from a variety of sources, as discussed above. For example, the user may attempt to launch a new application from a start menu or icon.”). Thus, Hassan teaches users launching (or attempting to launch) an application by selecting the particular icon for that application. *Id.* A POSITA would have understood this because that is how icons have been used, since at least the 1990s, to launch applications PCs and mobile devices such as mobile phones. Indeed, Hassan teaches that users initiate task changes by clicking on buttons. *Id.*, ¶¶34-36 (“To change the active application or window, a user may initiate a task change request. By way of example, in a Microsoft Windows™ operating system environment, a task change may be requested through a mouse click upon a button on a task bar showing the

available open windows. ...Within the displayed message, selection of a ‘reply’ or ‘forward’ button or menu item launches yet another window for composing the reply or forward message.”), ¶¶52-53, ¶59.

52. Second, for icons within an application, Hassan enables users to request or initiate an operation within an application (e.g., e-mail application) by clicking the icon for the operation, while the device is in locked mode. *E.g.*, Ex-1004, ¶¶70-72, ¶¶35-36, Abstract; *supra* §IV.A.3[1b]/[11b]/[17b] (explaining how Hassan teaches icons associated with operations). For example, Hassan teaches icons associated with a set of operations (e.g., forwarding, replying to, and opening messages) within an application (e.g., e-mail application) while the device is in locked mode. *Id.* Based on Hassan’s teachings, users select the icon associated with the operation by clicking the icon in order to request or initiate the operation. Ex-1004, ¶¶70-72, ¶36, Abstract. Thus, Hassan enables users to select an icon to request/initiate (or attempt to request/initiate) the associated operation. A POSITA would have understood this because users have clicked on UI buttons since early Mac and Windows computers in the 1990s, as Hassan teaches (*id.*, ¶34). Hassan teaches that, “in an e-mail application, a user may request a certain function like forwarding a message, replying to a message, opening a message, through an appropriate user input (like clicking on a designated icon).” *Id.*, ¶36, ¶72 (“The user may request or initiate the operation or task by selecting a menu item, clicking

an icon associated with the operation, or otherwise providing user input to trigger the operation.”).

53. Additionally, allowing users to select icons would have been obvious to a POSITA based on Hassan’s teachings because the use of selectable icons for launching applications or executing functions within applications had been extremely well known for years or even decades. Even early graphical user interfaces (GUIs) of the 1990s included this functionality. For example, the 1992 Macintosh Human Interface guidelines have an entire 33-page chapter on icons. Ex-1011, 000025-57. It explains that “[i]cons are graphic representations of objects such as documents, storage media, folders, applications, and the Trash. Icons look like their real-world counterparts wherever possible. People can select, open, move, copy, and throw away icons. Figure 8-1 shows some icons displayed in the Finder.” Ex-1011, 000026. Figure 8-1 shows application icons such as TeachText 7.0 as well as function icons such as the Apple icon in the upper left corner of the screen.

CHAPTER 8

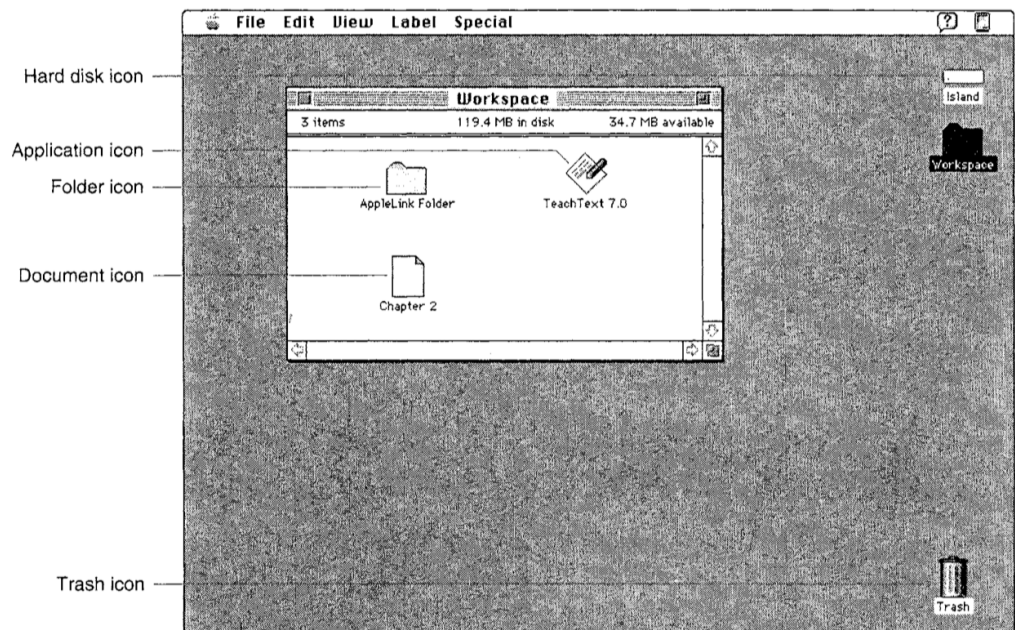
Icons



This chapter describes icons, their appearance, and their use in the Macintosh interface. It presents information on how to design icons and general guidelines for designing icons of different sizes and bit depths. This chapter also describes how to customize the standard icons you can provide for your products.

Icons are graphic representations of objects such as documents, storage media, folders, applications, and the Trash. Icons look like their real-world counterparts whenever possible. People can select, open, move, copy, and throw away icons. Figure 8-1 shows some icons displayed in the Finder.

Figure 8-1 Common icons



Ex-1011, 000026, Fig. 8-1. The Apple menu in the upper left corner of the screen represents a selectable menu option that is part of the Finder application as described in that same document:

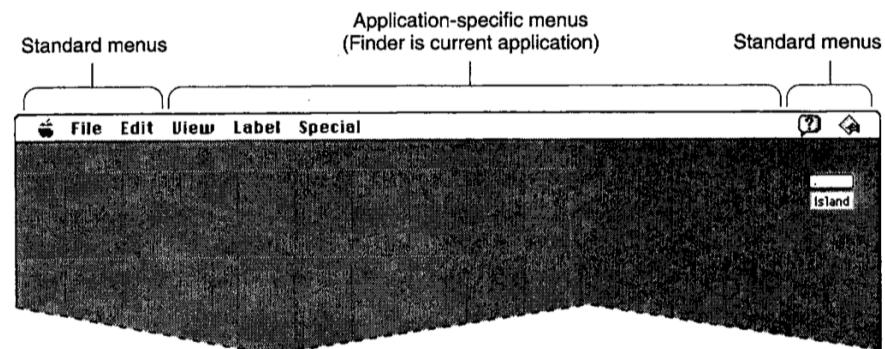
The Menu Bar

The **menu bar** extends across the top of the screen and contains words and icons that serve as the title of each menu. It should be visible and always available to use. Nothing should ever appear on top of the menu bar or obscure it from view. The menu bar should always contain the standard menus—the Apple menu, the File menu, the Edit menu, the Help menu, and the Application menu. The Keyboard menu is an optional standard menu that appears when the user installs a script system other than the Roman Script System. The titles of the standard menus never change. (The titles of the Apple, Help, Keyboard, and Application menus are icons rather than words.) The standard menus are described in the section “Standard Macintosh Menus,” beginning on page 98.

CHAPTER 4

Menus

Figure 4-2 A menu bar



The width of the menu bar depends on the screen size of the user's monitor. Figure 4-3 shows menu bars of three different widths and how much space is used by the Finder's menu titles on each screen.

Ex-1011, 000023-24.

54. Therefore, by 2007, a POSITA with the given level of education and experience (*supra* §II) would have grown up with user selection of icons as a basic, nearly universal way of interacting with computers, closely associated with the

basic use of a mouse, touchpad, or touchscreen, and would have understood Hassan to be teaching this type of user selection based on its description of user interactions with the device. *See* Ex-1005, ¶29 (“Note, the example of FIG. 2A provides some additional information as for example when a particular icon or element 29 is selected.”), Figs. 2-3, ¶36 (“In this first example of a means for implementation of the present invention, a generally highlighted area 36 is shown which provides, only in direct relationship to a specific item, here item or MP3 file 32, an expanded display of a multifunctional set 38 of features, here, operational icons, e.g., icons 38 *a*, 38 *b*, 38 *c* and 38 *d*. These operational icons 38 in being operational are thus selectable as well.”), ¶44 (“Toolbox items can be selected by pressing select key, see the gleaming phone icon 48 of FIG. 5C[.]”), ¶49 (“As shown by the gleaming icon 73 in FIG. 9B, the user selects a function (here, an exemplar save-function).”).

55. Hassan teaches enabling a user to select the object while the device is in the locked mode **in order to determine a state, function or utility of the application**. First, as explained above, Hassan teaches enabling a user to select the icon for an application to perform operations such as launching an application when the user requests a task change. Ex-1004, ¶¶59-61. Hassan teaches that, upon receiving a task change request, the device evaluates whether it is in a locked state, if not, the device implements the task change; if yes, the task change request is

evaluated to determine whether it is a permitted task. Ex-1004, ¶60. “If the task change is a permitted task,” the device then “proceeds [] to implement the requested task change.” *Id.* “If the task is not permitted,” the device does not implement the task change. Ex-1004, ¶61. Instead, the device turns to further processes, including, for example, displaying an authorization window to the user for inputting a password or other authorization data. *Id.* Hassan teaches numerous ways where users select an object to initiate a task, including for example “a button on a task bar” or other clickable icons. Ex-1004, ¶¶34-36, ¶¶52-53, ¶59. Hassan therefore teaches the user selecting an icon to determine a state of the related application (e.g., whether launching the application is a permitted task or not during locked mode, whether the application is launchable or not), a function of the related application (e.g., what functions of the application are permitted or not during locked mode), and a utility of the related application (e.g., what can the application be used for during locked mode). Hassan thus teaches that a user selects the icon relating to an application in order to determine a state, function, or utility of the application, for example, in order to determine whether an application can be launched to perform the requested task.

56. Patent Owner’s district court positions regarding how the claims can be satisfied confirm that Hassan’s teachings fall within the scope of this limitation. For example, Patent Owner broadly alleges that this limitation is satisfied by icons

in the Android UI where the user can interact with the icon during the alleged locked mode, including a range of objects for changing Wi-Fi and Bluetooth operation. Ex-1009, 000007-10 (accusing possible user interactions for “pressing or tapping an object, such as the object related to a wi-fi application or Bluetooth application, determines the state, function, or utility of that related application without fully activating the application.”).

57. Second, for icons within an application (explained above), Hassan teaches that users may request or initiate an operation within an application (e.g., e-mail application) by clicking the icon associated with the operation when the device is in locked state. *E.g.*, Ex-1004, ¶¶70-72, ¶36, Abstract. Hassan teaches that operations within the e-mail application include permitted or impermissible ones during the locked state, such as forwarding, replying to, and opening messages. *Id.* These operations represent the functions and/or utilities of the application. Hassan therefore teaches that a user selects the icon for an operation in order to determine a function of the application (e.g., whether or not the application provides the functionality of forwarding, replying to, or opening e-mail messages during locked mode), and a utility of the application (e.g., whether or not the application can be used to forward, reply to, or open email messages during locked mode). In addition, Hassan teaches that a user selects such icons in order to determine a state of the application, for example, whether the associated tasks can

be performed in locked mode. As explained above, the Patent Owner's district court positions confirm that Hassan's teachings fall within the scope of the challenged claims.

58. Hassan further teaches enabling a user to select the object in order to determine a state, function or utility of the application **without fully activating the application while the device remains in the locked mode**. For example, Hassan teaches that the application (e.g., e-mail application) is not fully activated because users may only access a limited set of "permitted" tasks or operations within the application, rather than the full set of operations, as explained above. Ex-1004, ¶¶70-72, ¶38. Since the application cannot perform its full set of operations, it is not fully activated. *See id.*

59. As Hassan explains, this facilitates the sharing of devices with other users without compromising security because it allows the user to temporarily share the device with other people. Hassan addresses situations where a user "share[s] the device with colleagues for the purpose of allowing them to view a particular e-mail message" without allowing them full access to the email application. Ex-1004, Abstract, ¶¶1-3. Hassan also addresses the use case where "the device owner may only wish to share the device with children for the purpose of gaming without allowing them access to e-mail or other messaging applications." *Id.* "In yet another example, the device owner may wish to share the

device with a colleague to make a voice call without allowing access to e-mail or other applications.” *Id.* In each of these use cases—which are the scenarios that Hassan seeks to address—the sharing of the device and the activation of the application in locked mode is only temporary. *Id.* Hassan teaches that, when a user requests or initiates a task while the device is in the locked state, the associated application is only temporarily activated, without full access, to allow the user to execute the requested task if it is permitted; otherwise, if not permitted, the device would need to be unlocked to perform the task. Ex-1004, ¶¶70-72. Therefore, Hassan teaches that the application (e.g., email) is not fully activated while the device remains in locked mode; instead, it is temporarily activated with limited functionality, which serves the underlying purpose of Hassan. Ex-1004, ¶¶1-3, ¶¶13-14, Abstract.

60. For the reasons set forth above, Hassan teaches limitations [1c]/[11c]/[17c]. Grounds 1-2 therefore teach limitations [1c]/[11c]/[17c].

4. Dependent Claims 2 and 13

2. The apparatus as claimed in claim 1, wherein the computer program code is configured, working with the at least one processor, to cause at least the following to be performed:
13. The method as claimed in claim 11, further comprising:
responding to user selection of the object by illuminating or highlighting the object.

61. Ground 2 teaches claims 2 and 13. Hassan teaches the apparatus of claim 1, including computer program code that is configured, working with the at least one processor, to cause the steps described in this Declaration to be performed, and the method of claim 11. *Supra* §IV.A.3. The combination of Hassan with Nirhamo (Ground 2) further teaches the additional limitations of claims 2 and 13.

62. As explained above, Hassan teaches the user selecting an object related to an application. *Supra* §§IV.A.3[1b]-[1c]. Nirhamo further teaches **responding to user selection of the object by illuminating or highlighting the object**. Nirhamo “relates to improved presentation, navigation, selection and/or operation options for portable communication devices, particularly, user interface options involved on the display screens thereof.” Ex-1005, ¶1. Nirhamo discloses device operations, including highlighting items on the display when they are selected. Ex-1005, Abstract, ¶4, ¶¶29-31, ¶¶36-39, ¶41, ¶51, ¶71. For example, Nirhamo teaches that “when a particular icon or element[] is selected[,]

[t]his element[] may then be highlighted in some way (as here by a distinctively colored box framed therearound, although other means are also usable herewith).” Ex-1005, ¶29, ¶51 (“Generally, focused-upon items are shown here marked with dotted backgrounds (though these could be highlighted otherwise, e.g., by being brightened or gleamed relative to other selection alternatives or by being presented in a distinctive colour, or other style, inter alia). Up and down keys of a joystick (or other cursor movement implementation such as a four or eight way button) can be used to select the establishment of a focus or highlighting on an item, e.g., ‘Item 1’ element 81, ‘Item 2’ element 82, and/or ‘Item 3’ element 83. ... In these embodiments, the ‘items’ 81, 82, 83 may be considered as either selectable items or features as these terms are used throughout.”).

63. A POSITA would have been motivated to apply Nirhamo’s user-selection teachings with Hassan because Hassan teaches the user selecting icons to determine whether or not the associated application or operation was permitted. *Supra* §IV.A.3[1c]. It was customary for device interfaces to highlight the user’s selection so that the user could see and confirm what was selected. For example, Hassan teaches the user selecting icons to launch applications or initiate operations within an application. Ex-1004, ¶59, ¶¶70-72, ¶36; *supra* §IV.A.3[1c]. For this to happen, it would have been obvious to highlight the selected icon, so that the user can see which application or operation was currently selected (e.g., for forwarding,

replying to, or opening a message). This was a common, universal feature in device interfaces since the early GUIs of the 1990s. If the user's selection was not highlighted in some fashion, it would have been difficult for the user to navigate the device interface (e.g., computer interface), as there would not be a clear way for the user to discern whether the correct object had been selected.

64. Highlighting the user's selection would not have changed the principle of operation of either reference. Indeed, it would have furthered Hassan's teachings by facilitating the user's selection of icons. A POSITA would have had a reasonable expectation of success in combining Hassan with Nirhamo's teachings for highlighting user selections because the combination would have been a common, routine, and expected operation that is suggested by Hassan's teaching of user selection. Highlighting user selection was implied by Hassan because that is how device interfaces (e.g., computer interfaces) had operated for many years. Ex-1011, 000023-57, Fig. 8-1.

65. The combination of Hassan with Nirhamo therefore teaches responding to user selection of the object by illuminating or highlighting the object. Ground 2 thus teaches claims 2 and 13.

5. Dependent Claims 3 and 14

- 3. The apparatus as claimed in claim 2,**
- 14. The method as claimed in claim 11,**
- wherein a movable object is provided to enable the user to select the object.**

66. Ground 2 teaches claims 3 and 14. The combination teaches the apparatus of claim 2, and the method of claim 11. *Supra* §§IV.A.3, IV.A.4. The combination of Hassan with Nirhamo (Ground 2) further teaches the additional limitations of claims 2 and 13.

67. The combination teaches that a movable object (e.g., cursor) is provided to enable the user to select the object. As explained above, Hassan teaches that a user selects an object related to an application. *Supra* §IV.A.3. Hassan further explains that the user may select the object using a movable object such as a cursor controlled by a mouse. *E.g.*, Ex-1004, ¶34 (“To change the active application or window, a user may initiate a task change request. By way of example, in a Microsoft WindowsTM operating system environment, a task change may be requested through *a mouse click upon a button on a task bar showing the available open windows.*”), ¶35 (“Similarly, within an e-mail messaging program, clicking on a message in an ‘inbox’ view launches a new window for displaying the selected message.”). When users control a mouse, it moves a cursor on the screen. A cursor is a movable object because it is an object moved based on user

input, to manipulate its position so that the user can navigate through the user interface and perform operations such as selecting items on the display.

68. Nirhamo reinforces Hassan's teachings. Nirhamo similarly discloses using movable objects such as a cursor for user operations (e.g., selecting items in the display) and confirms that this was known in the art. Ex-1005, ¶39 (“[T]he user can move an emphasized or otherwise highlighted cursor or visual selection representation (here shown by bolding and/or the darker coloring of the play/pause button 38 a) to select the desired functional operation to be performed for the selected item 32[.]”), ¶51 (“Up and down keys of a joystick (or other cursor movement implementation such as a four or eight way button) can be used to select the establishment of a focus or highlighting on an item, e.g., ‘Item 1’ element 81, ‘Item 2’ element 82, and/or ‘Item 3’ element 83.”), ¶23 (“[T]he keypad 2 has a first group 7 of data entry buttons or keys as alphanumeric keys, two softkeys 8, and a scroll-key 10 (up/down and/or right/left and/or any combination thereof) for moving a cursor in the display 3. An alternative hereto may be a four-way button, an eight-way button or a joystick, track ball, roller or other cursor controller [.]”), ¶38 (“The operator or user of the phone can then initiate or otherwise change the desired function to be used directly in the highlighted area 36 using phone cursor control keys, such as for example, an arrow key or keys, see multidirectional

key 10 in FIG. 1 (alternative multidirectional keys, joysticks, rollers etc. or individual right and left or up and down keys may otherwise be used as well).”).

69. A POSITA would have been motivated to combine Nirhamo’s cursor teachings with Hassan because it provides additional context for the known user interface taught by Hassan. Specifically, Nirhamo describes additional details of pointing, clicking, etc., which Hassan already suggests because Hassan teaches users pointing and clicking with a mouse. Using input/point devices (e.g., a cursor) to facilitate user interactions has been known for decades and was widely applied across different operating systems and software applications. *See* Ex-1007, ¶29 (“[S]ome of the user interfaces and associated processes may be applied to other devices, such as personal computers and laptop computers, which may include one or more other physical user-interface devices, such as a click wheel, a physical keyboard, a mouse and/or a joystick.”), ¶40 (“[I]nput controller(s) 160 may be coupled to any (or none) of the following: a keyboard, infrared port, USB port, and a pointer device Such as a mouse.”); Ex-1010, 3:22-43, 8:13-26, 16:32-47 (“In a simple graphics model, object may be selected by clicking directly on the object with the mouse or dragging a selection box across several objects using a mouse tracker.”), 37:51-38:3 (“A graphical user interface is manipulated by moving a mouse, clicking on objects to select them, dragging objects to move or copy then, and double-clicking to open them.”), 41:8-20 (“Users of an object oriented

operating system's graphical user interface often move a mouse, click on objects to select them, drag objects to move or copy them, and double-click to open an object."), Fig. 1A.

70. Therefore, the combination teaches providing a movable object (e.g., cursor) in the user interface to enable the user to select the object. Ground 2 thus teaches claims 3 and 14.

6. Dependent Claim 4

4. The apparatus as claimed in claim 2, wherein the object is highlighted or illuminated to a lesser extent than that associated with a standard mode setting.
--

71. Ground 2 teaches claim 4. The combination teaches the apparatus of claim 2. *Supra* §IV.A.4. The combination of Hassan with Nirhamo (Ground 2) further teaches the additional limitations of claim 4.

72. As explained above, the combination teaches responding to user selection of the object by illuminating or highlighting the object. *Supra* §IV.A.4. The combination further teaches that **the object is highlighted or illuminated to a lesser extent than that associated with a standard mode setting**. As an initial matter, Nirhamo teaches illuminating or highlighting an item by changing its brightness. Ex-1005, ¶51 ("Generally, focused-upon items are shown here marked with dotted backgrounds (though these could be highlighted otherwise, e.g., by being brightened or gleamed relative to other selection alternatives or by being

presented in a distinctive colour, or other style, inter alia).”), ¶43 (“In such a single-item listing, the display 40 including a list 41 as shown first in FIG. 5A, a focus 43 (by gleaming, color or brightness or other highlight change) is placed on an item 42 in the list 41.”), ¶39 (“Thus, in this example, the user can move an emphasized or otherwise highlighted cursor or visual selection representation (here shown by bolding and/or the darker coloring of the play/pause button 38 a) to select the desired functional operation to be performed for the selected item 32[.]”).

73. Nirhamo further discloses dimming items that are unavailable or otherwise incompatible with an operation. Ex-1005, ¶36, ¶¶53-54 (“... Here only the currently focused item is de-emphasized or dimmed when the action indication is focused on ‘Cancel.’ See ‘Item 2’ element number 82 b. This de-emphasis shows the direct relationship of the action ‘Cancel’ to the highlighted item 82 b, specifically, that the action is not applicable to the item.”), ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C. These items have “de-emphasized” representations that contrast with available items to indicate their unavailability. Ex-1005, ¶¶53-54, ¶36 (“Note, the other selectable items, e.g. items 33, 34, 35, inter alia, have contrasting non-highlighted representations.”). Being dimmed refers to a darker, i.e., reduced brightness level, meaning a dimmed object is illuminated to a lesser extent. *See,*

e.g., Ex-1012, 000003 (explaining “dim” and “dimmed”); Ex-1013, 000003 (explaining “dim” and “dimmed”).

74. Hassan teaches that some icons are unavailable when the device is locked because the associated applications are not permitted or the associated operations (within an application) are not permitted during the locked state. *Supra* §IV.A.3[1c]. A POSITA would have been motivated to apply Nirhamo’s teachings by dimming these unavailable icons because it indicates to the user that the application cannot be launched or the operation is not permitted during locked mode. *See* Ex-1005, ¶36, ¶¶53-57, Figs. 10B-10C, Figs. 11A-11C. This would have provided a simple, intuitive way to implement Hassan’s teachings and indicate to the user which icons were not permitted, consistent with general computer interfaces. Indeed, it was common in computer interfaces and expected for items to be greyed out or dimmed to indicate unavailability, for example in Microsoft Windows, which Hassan cites. Ex-1004, ¶22, ¶34. A POSITA would have had a reasonable expectation of success in combining these teachings because it was simply a matter of dimming an icon on the computer display, and reducing brightness was a common and simple setting in computer devices by 2007.

75. Therefore, Hassan teaches that some icons are unavailable during the locked state (*supra* §IV.A.3), and the combination with Nirhamo teaches dimming those icons, meaning they are highlighted or illuminated to a lesser extent than a

standard mode of operation, when all applications and operations are available. Ex-1005, ¶36, ¶¶53-54, ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C. In other words, when the device is in a standard, unlocked mode of operation, those applications or operations are available and no longer dimmed. When the user selects those icons to determine whether or not they are permitted, Nirhamo teaches highlighting items to indicate the user selection, for example with “a distinctively colorized box framed” as explained for claim 2 (Ex-1005, ¶29, ¶51; *supra* §IV.A.4), and also dimming and de-emphasizing the item to indicate its unavailability (Ex-1005, ¶36, ¶¶53-54, ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C).

76. Ground 2 therefore teaches claim 4.

7. Dependent Claims 5 and 15

5. The apparatus as claimed in claim 1,
15. The method as claimed in claim 11,
wherein the object represents an application that is present or
available in the device.

77. Grounds 1-2 teach claims 5 and 15. Hassan teaches the apparatus of claim 1, and the method of claim 11. *Supra* §IV.A.3. Hassan further teaches the additional limitations of claims 5 and 15.

78. As explained above, Hassan teaches causing a display of a device to display an object relating to an application and that a user selects the object to determine a state, function or utility of the application.

Supra §§IV.A.3[1b]/[11b]/[17b]-[1c]/[11c]/[17c]. Hassan further teaches that **the object represents an application that is present or available in the device**. For example, Hassan teaches that a user launches applications on the device by selecting the icon representing the application. *Supra* §IV.A.3; Ex-1004, ¶59. In addition, the user clicks an icon for an operation to request or initiate that operation within an application that is present or available in the device. *Supra* §IV.A.3; Ex-1004, ¶¶70-72, ¶36, Abstract. The icon, either representing the application or the operation within the application, presents an application that is present or available in the device. Hassan also teaches “[u]ser input and interaction is with the active window and, therefore, its associated application is the active application.” Ex-1004, ¶34. Grounds 1-2 therefore teach claims 5 and 15.

8. Dependent Claim 6

6. The apparatus as claimed in claim 1, wherein the computer program code is configured, working with the at least one processor, to cause at least the following to be performed: causing the display to display another object, relating to another application, when the device is in a locked mode.

79. Grounds 1-2 teach claim 6. Hassan teaches the apparatus of claim 1. *Supra* §IV.A.3. Hassan further teaches the additional limitations of claim 6.

80. Hassan teaches **causing the display to display another object** (e.g., icons or windows), **relating to another application, when the device is in a locked mode**. As explained above, Hassan teaches representing an application by

an icon and that a user selects the icon to launch the application when the device is in the locked state, including an email application. *Supra* §IV.A.3. Hassan further teaches the device having multiple applications. *E.g.*, Ex-1004, ¶27 (“Operating system software 52 and various software applications 54 used by the microprocessor 38 are, in one example embodiment, stored in a persistent store such as flash memory 24 or similar storage element.”), ¶28 (“A predetermined set of software applications 54 which control basic device operations, including data and voice communication applications for example, will normally be installed on the device 10 during manufacture.”), Fig. 2 (showing a mobile device with various applications implemented, color added):

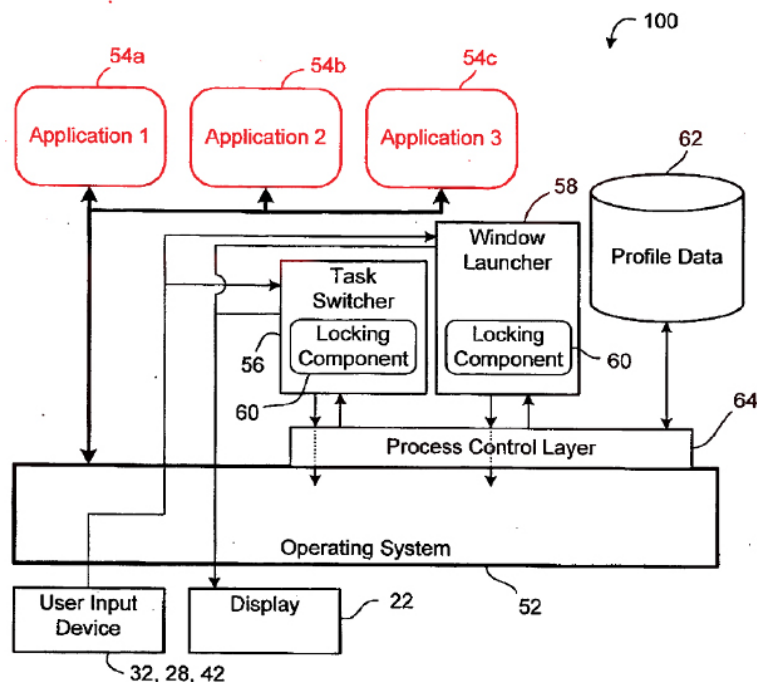


FIG. 2

Hassan explains that “two application programs may be open in two separate windows within the display, an active window being layered on top of an inactive window. ... User input and interaction is with the active window and, therefore, its associated application is the active application. To change the active application or window, a user may initiate a task change request.” Ex-1004, ¶34.

81. Hassan teaches icons for launching applications. Ex-1004, ¶59, ¶¶70-72, ¶36. This was explained above for claim 1. *Supra* §IV.A.3[1b]. One of those icons, e.g., for the email application, is the icon recited in claim 1. *Id.* Any of the other icons is another object, relating to another application, that Hassan’s device causes to be displayed (e.g., through the “Operating System”) so the user can interact with and potentially launch the application (e.g., via the “User Input Device,” “Task Switcher,” and “Window Launcher”). Ex-1004, ¶36, ¶¶43-44, ¶59, ¶64, ¶¶70-72, Fig. 2. Thus, Hassan teaches displaying another object, relating to another application, when the device is in a locked mode. Grounds 1-2 therefore teach claim 6.

9. Dependent Claim 7

7. The apparatus as claimed in claim 6, wherein the another object represents an application that is present or available in the device.

82. Grounds 1-2 teach claim 7. Hassan teaches the apparatus of claim 6. *Supra* §IV.A.8. Hassan further teaches the additional limitations of claim 7.

83. As explained above, Hassan causes the display to display another object relating to another application (e.g., an icon representing another application/an operation within another application), when the device is in a locked state. *Supra* §IV.A.8. Hassan teaches that the another object (e.g., an icon or a window) represents an application that is present or available in the device, and that the user may select the object to launch the application. Ex-1004, ¶¶59, ¶¶70-72, ¶34, ¶36. Therefore, Hassan teaches the another object represents an application that is present or available in the device. Grounds 1-2 therefore teach claim 7.

10. Dependent Claim 8

8. The apparatus as claimed in claim 6, wherein the another object indicates that the another application is not activatable when the device is in the locked mode.

84. Ground 2 teaches claim 8. Hassan teaches the apparatus of claim 6. *Supra* §IV.A.8. Ground 2 further teaches the additional limitations of claim 8.

85. The combination of Hassan and Nirhamo teaches that **the another object indicates that the another application is not activatable when the device is in the locked mode**. As explained above, the combination teaches that an object in the display (e.g., dimmed icon) indicates that the object is unavailable or otherwise incompatible with an intended action/operation. *Supra* §IV.A.6; Ex-1005, ¶53 (“Items in the list 86, see particularly items 81 *a*, 82 *a*, and 83 *a*, are de-

emphasized or dimmed in FIG. 10B, as indicated here by the distinctive less bold font, so that the user knows they are not part of the possible or intended “Cancel” action suggested here by the action focus/indication on the “Cancel” action element 85 *a*.”), ¶54, ¶¶56-57 (“See for example, the display 90 of FIG. 11A which shows three items 91, 92 and 93, where however, the third such item 93, ‘item 3,’ is shown dimmed (indicated by the distinctive, less-bold font). This is dimmed when the first action in this example, here ‘Act 1,’ element 94, is highlighted (see the dotted background thereof), thereby *indicating that item 93 is unavailable for or otherwise incompatible with operation by ‘Act 1’ 94*. ... FIG. 11B shows in a display 90 *a* an alternative situation when for example the ‘Act2’ element 95 *a* has been selected and then corresponding unavailable items 91 *a* and 92 *a* (‘items 1 and 2[’]) are dimmed.”), ¶60, Figs. 10B-10C, Figs. 11A-11C.

86. Hassan teaches that “impermissible applications” cannot be activated when the device is in the locked state. Ex-1004, ¶39 (“In some embodiments, the device 10 may further include stored profile data 62. The profile data 62 may define the permitted and/or impermissible tasks associated with a locked state. In one embodiment, the profile data 62 may include permitted application programs. For example, the profile data 62 may list a phone application program and/or various games application programs but may not include an e-mail or messaging application, word processing applications, and/or systems-level applications such

as control panel or uninstall facilities. In a locked state the user can only activate one of the permitted programs included in the profile data 62. Those skilled in the art will appreciate that rather than indicating permitted applications, the profile data 62 may indicate the impermissible applications.”). Therefore, based on the combined teachings, a dimmed icon or window indicates that the another application (e.g., impermissible application) is not activatable when the device is in the locked mode. Ground 2 therefore teaches claim 8.

11. Dependent Claims 9 and 16

- 9. The apparatus as claimed in claim 1,**
16. The method as claimed in claim 11,
wherein the display is a touch screen display, and enabling a user to select the object comprises enabling the user to select the object using the touch screen display.

87. Ground 2 teaches claims 9 and 16. Hassan teaches the apparatus of claim 1, and the method of claim 11. *Supra* §IV.A.3. The combination of Hassan with Nirhamo (Ground 2) further teaches the additional limitations of claims 9 and 16. For example, Nirhamo teaches that touch screen could be used to implement the operations it discloses. Ex-1005, ¶23 (“Touch screen functionality could also be used.”). Moreover, using touch screen displays to facilitate user interactions (e.g., selecting objects displayed on the touch screen) was widely known in the art well before the timeframe of the ’741 patent. *See* Ex-1007, Abstract, ¶¶7-11, ¶77 (“The touchscreen may display one or more graphics. ... [A] user may select one

or more of the graphics by making contact or touching the graphics, for example, with one or more fingers 202 (not drawn to scale in the figure) or a stylus (not shown in the figure).”), ¶119, ¶125 (“In some embodiments, the user interface includes a finish icon 510 having text such as ‘done,’ ‘okay,’ or ‘save.’ A user selection of this icon causes the portable device to complete the user configuration process and return to the user interface[.]”), ¶126, Fig. 2; Ex-1010, 3:22-43.

88. It would have been obvious to modify Hassan based on Nirhamo’s teachings such that the display of the device is a touch screen display. A POSITA would have understood that devices with touch screen displays allow users to select objects using the touch screen.

89. A POSITA would have been motivated to combine Nirhamo’s touchscreen teachings to Hassan for several reasons. Hassan teaches an application running on a mobile device. Ex-1004, ¶¶5-11, ¶20 (“[T]he present application may be embodied within a variety of user devices, including mobile devices, handheld devices, mobile telephones, personal digital assistants (PDAs) and other such devices.”). Touchscreens were common features on mobile devices by 2007, and Nirhamo teaches a mobile device with a touchscreen. Ex-1005, ¶23, Fig. 1A. Therefore, the inclusion of a touchscreen with Hassan’s teachings would have used a known technique to improve similar mobile devices in the same way.

90. Moreover, the combination would have had several advantages.

Touchscreens offered an intuitive user interface because users could simply touch the screen to interact with the computer directly. For that reason, touchscreens were often touted as a marketing feature in laptops. Therefore, a POSITA would have had ample motivation to combine Nirhamo's teachings with Hassan.

91. The combination would not have changed the principle of operation for Hassan or Nirhamo. The combination simply adds a touchscreen to Hassan's mobile device, which was a common peripheral that adds an additional way for the user to interact with the device but does not alter the manner in which Hassan's teachings were applied to a mobile device. Nor would it have changed Nirhamo's principle of operation since the touchscreen still would have operated as a touchscreen. A POSITA would have had a reasonable expectation of success in combining these teachings because it was simply a matter of replacing the display with a touch screen display, and touch screen display was common and widely used by 2007.

92. Ground 2 therefore teaches claims 9 and 16.

12. Dependent Claim 10

10. The apparatus as claimed in claim 1, wherein the apparatus is the device.
--

93. Grounds 1-2 teach claim 10. Hassan teaches the apparatus of claim 1.

Supra §IV.A.3. As explained above, Hassan teaches its mobile device as an

apparatus that comprises the processor and memory that cause the operations described in limitations [1b]/[11b]/[17b]-[1c]/[11c]/[17c] to be performed. *Supra* §IV.A.3. Hassan therefore teaches that the apparatus is the device, i.e., the mobile device. Grounds 1-2 therefore teach claim 10.

13. Dependent Claims 12 and 18

12. The method as claimed in claim 11,
18. The non-transitory computer readable medium as claimed in claim 17,
wherein the object is a dimmed icon.

94. Ground 2 teaches claims 12 and 18. Hassan teaches the method of claim 11, and the non-transitory computer readable medium of claim 17. *Supra* §IV.A.3. The combination of Hassan with Nirhamo (Ground 2) further teaches that **the object is a dimmed icon**. As explained above, Hassan teaches that some icons are unavailable during the locked state (*supra* §IV.A.3), and the combination with Nirhamo teaches dimming icons when the associated applications or operations are unavailable. *Supra* §IV.A.6; Ex-1005, ¶36, ¶¶53-54, ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C. The combination therefore teaches that the object is a dimmed icon, and thus teaches claims 12 and 18.

14. Dependent Claim 19

19. The apparatus as claimed in claim 17, wherein the object is an icon.

95. Grounds 1-2 teach claim 19. Hassan teaches the apparatus of claim 17. *Supra* §IV.A.3. As explained above, Hassan teaches that the object is an icon that represents an application or an operation within an application. *Supra* §IV.A.3; Ex-1004, ¶59, ¶¶70-72. Hassan therefore teaches that the object is an icon. Grounds 1-2 thus teach claim 19.

15. Dependent Claim 20

20. The apparatus as claimed in claim 19, wherein the icon is a dimmed icon.

96. Ground 2 teaches claim 20. As explained above, Hassan teaches the apparatus as claimed in claim 19 and that the object is an icon. *Supra* §IV.A.14. As explained above, the combination of Hassan with Nirhamo (Ground 2) teaches a dimmed icon. *Supra* §IV.A.13. Ground 2 therefore teaches claim 20.

B. Grounds 3-4

97. It is my opinion that a POSITA would have found the Challenged Claims obvious based on Becker (Ground 3). Several dependent limitations recite simple limitations including, for example, highlighting or illuminating an object (claims 2, 4), using a movable object (claim 3), using a touch screen display (claim 9), and displaying a dimmed icon (claim 12). Those limitations are taught by the

combination of Becker and Nirhamo (Ground 4), which together render claims 1-20 obvious.

98. Becker teaches limited user access to a telephone application while the computer is in a locked screen saver mode. *Infra* §IV.B.1. A POSITA would have been motivated to combine Becker with Nirhamo's teachings for, e.g., dimming items that are unavailable or otherwise incompatible with an operation, and using a touch screen display. *Infra* §§IV.B.5 (explaining application of Nirhamo's dimming teachings to Becker), IV.B.10 (explaining application of Nirhamo's touch screen teachings to Becker). These had been standard features in computers for many years before 2007. Therefore, the combination of Ground 4 combines known, prior art elements according to known methods to yield predictable results (e.g., a computer touchscreen that dims objects on the display during a screen saver).

99. The motivation and reasonable expectation of success in combining these prior art elements is explained in more detail below. *See infra* §§IV.B. 3, 5, 10.

1. U.S. Patent No. 7,689,939 ("Becker") (Ex-1006)

100. I have reviewed the Becker reference. I understand that Becker was not cited or considered during prosecution of the '741 Patent, based primarily on

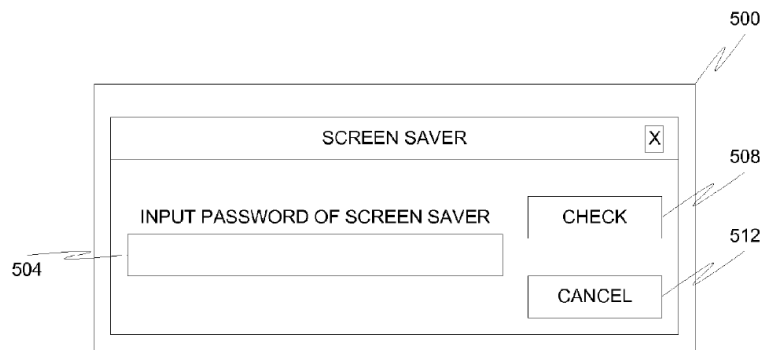
the fact that Becker is not cited on the face of the patent, nor have I seen the reference discussed in the prosecution history.

101. Becker was patented March 30, 2010, and was filed as application 10/882,889 on June 30, 2004. I understand Becker is prior art under at least pre-AIA §102(e) because it issued as a patent and was filed before the earliest possible filing date for the '741 patent, its provisional application's filing date of June 29, 2007.

102. Becker "relates generally to screen blankers and particularly to an application communicating with a user when a screen blanker is activated." Ex-1006, 1:6-8. "The invention is directed to a personal computer able to modify automatically a screen saver in response to the occurrence of one or more selected events." Ex-1006, 2:34-36. Becker's screen saver (also described as screen blanker) is a locked mode that allows limited user interaction with a defined set of applications, for example "for limited calling with specified call permissions," without fully activating them but otherwise prohibits access to normal applications. Ex-1006, 3:20-29 ("It can allow for limited calling with specified call permissions (or limited calling to specified destinations) while the PC is password protected."), 2:61-3:3 ("No other software application may be accessed while the display is in the screen saver mode."), 3:12-16 ("[N]o other software applications can be

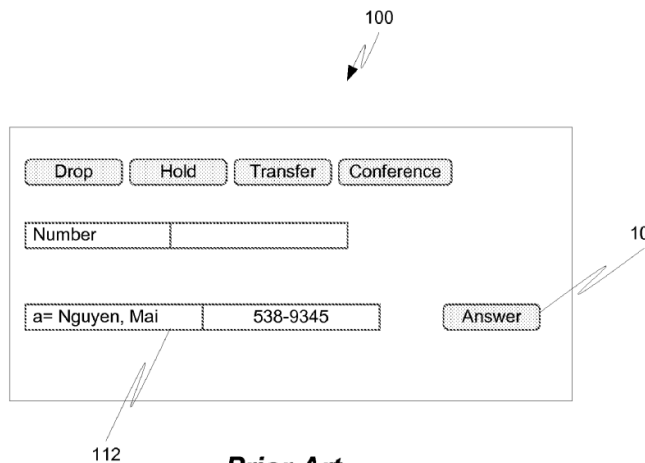
accessed while the password-protected screen saver is engaged”), 4:25-40, 5:51-56, 6:17-26, 6:48-60.

103. Becker incorporates conventional user interface features into its embodiments and teachings. This, for example, includes “a conventional screen saver display” (shown in Fig. 5) where the user must enter a password to unlock the computer. Ex-1006, 1:48-2:4, 3:50, Fig. 5:

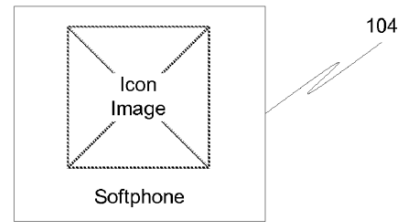


**Prior Art
Fig. 5**

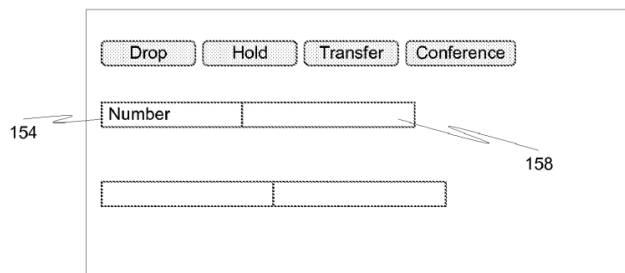
104. Becker’s teachings also include conventional user interface windows for a phone application. Ex-1006, 2:9-13, 3:41-42 (“FIGS. 1A-D depict various Graphical User Displays or GUI’s for a conventional softphone application[.]”), Figs. 1A-1D:



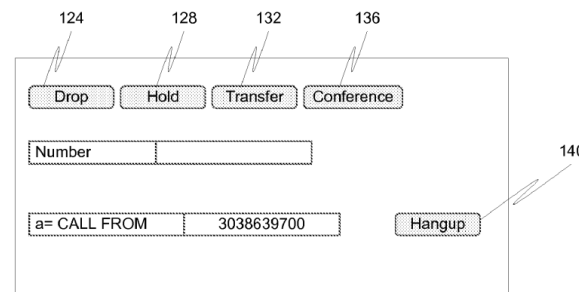
**Prior Art
Fig. 1A**



**Prior Art
Fig. 1B**



**Prior Art
Fig. 1C**



**Prior Art
Fig. 1D**

105. Becker allows the user to interact with these windows in screen saver mode. *E.g.*, Ex-1006, 2:9-13 (“A ringing call be answered even though the password protected screen blanker is activated.”).

106. Becker teaches the known operation of selecting an object (e.g., button in a GUI) related to application to determine a function or utility of the application when the device is in screen saver mode. For example, Becker enables a user to select buttons, including the “Answer” button, to determine a function of a communication application (e.g., the ability to answer, drop, hold, transfer, or

conference incoming calls), when the device is in a locked state. Ex-1006, 2:61-3:11, 6:48-7:17. Becker also enables a user to select buttons to determine a utility of the communication application (e.g., whether the application can be used to answer, drop, hold, transfer, or conference incoming calls during screen saver mode). Ex-1006, 5:1-24, 2:61-3:11, 6:48-7:17.

107. Therefore, Becker utilizes prior art GUI application windows for its teachings. Ex-1006, 1:29-47, 3:40-50, 4:25-5:47, 6:4-7:17, Figs. 1A-1D, Fig. 5. Therefore, these prior GUI windows are part of Becker's teachings. Additionally, a POSITA would have found it obvious and been motivated to combine Becker's embodiments with Becker's teachings of prior art GUI windows because Becker provides express motivation to do so. Moreover, the combination would have had advantages of using known GUI windows that a POSITA would have been familiar with, and providing quick access to certain operations, per Becker's teachings.

108. Becker is in the same field of endeavor as the '741 patent because it is directed to user interfaces. *See supra* §II; e.g., Ex-1006, 1:6-8, 2:34-36.

2. Independent Claims 1, 11, and 17

109. Beyond their preambles, these claims recite identical limitations and are therefore addressed together.

[1a]. An apparatus, comprising: at least one processor; and at least one memory storing computer program code configured, working with the at least one processor, to cause at least the following to be performed:

110. Becker implements its teachings with a PC that comprises a processor and memory that stores computer program code (e.g., software module). Ex-1006, Abstract, 4:25-4:40 (“As will be appreciated, the screen saver application 308 typically is software that may form part of the PC operating system, is stored in the PC’s hard disk, and is loaded in the main memory of the PC, when the PC is booted, to be executed by the control of the processor 312.”), 4:51-67 (“FIG. 3 depicts, in block diagram form, the structure of the software instructions stored in the memory 322 (or a computer readable medium) of the PC 216 and executed by the processor 312 to provide functions for the handset or for the headset and for the display device 316. The messages transmitted among the software elements or modules 300, 320, 304, 324, and 308 and hardware elements 316 and 328 are conveyed via a standard operating system not illustrated in FIG. 3. Drivers 320 and 324 provide the standard software interface to sound card 212 and LAN card 220, respectively. Such drivers are well-known to those skilled in the art.”), Fig. 3:

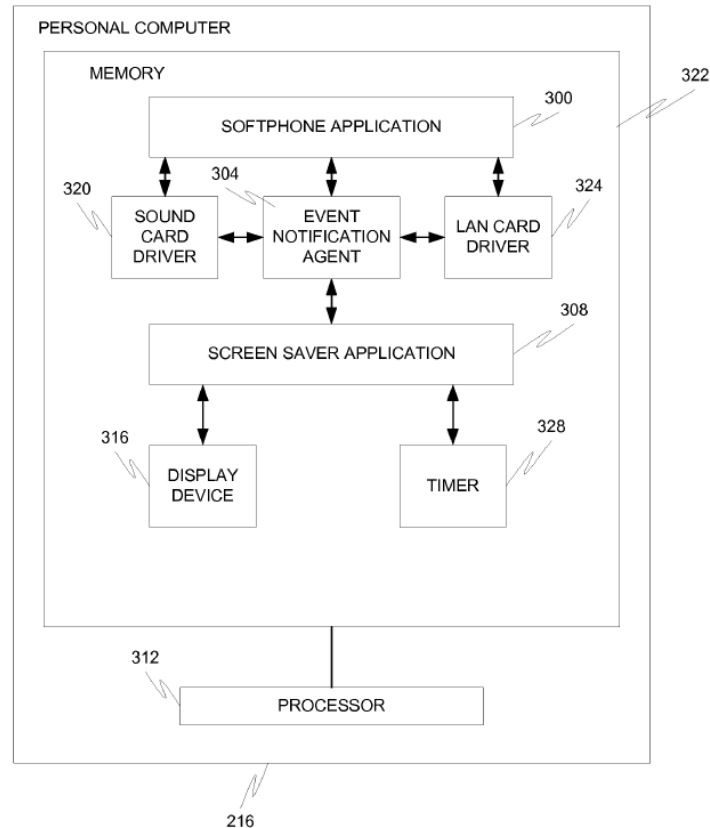


Fig. 3

Therefore, Becker teaches its PC as an apparatus that comprises such a processor and memory.

111. Becker's computer program code is configured, working with the processor, to cause its teachings to be performed, including for the below-described steps for [1b]-[1c] and claims 2-10. Ex-1006, 4:25-4:40, 4:51-67, Fig. 3. Additionally, a POSITA would have been knowledgeable about basic computer operation and understood that processors, such as those taught by Becker, execute computer program code in memory to cause the device to carry out its programmed functionality. That is how computers have operated for decades.

112. For the reasons set forth above, Grounds 3-4 teach the preamble and additional limitations of element [1a].

[11a]. A method, comprising:

113. Becker teaches a method, including operations as described for limitations [11b]-[11c] and claims 12-16. *E.g.*, Ex-1006, 2:40-3:11 (“In one embodiment, *a method* is provided that includes the steps of: (a) when the display is in a screen saver mode, receiving, from an application, at least part of which is executing on the personal computer, notification of the occurrence of an event; (b) determining if the event is a member of a defined set of events; and (c) when the event is a member of the defined set of events, displaying on the display (while the display is in the screen saver mode) image information associated with the event. ...”), 7:43-46. Limitations [11b]-[11c] are identical to limitations [1b]-[1c]. As explained below, Becker teaches operations described for limitations [11b]-[11c]. *Infra* §§IV.B.2[11b]-[11c].

114. For the reasons set forth above, Grounds 3-4 teach this preamble.

[17a]. A non-transitory computer readable medium storing computer program code that, when performed by at least one processor, causes at least the following to be performed:

115. Becker teaches implementing its methods using program code (e.g., software module) residing on computer readable storage medium (e.g., a PC’s

main memory) that is performed by a processor. Ex-1006, 4:25-4:40 (“As will be appreciated, the screen saver application 308 typically is software that may form part of the PC operating system, is stored in the PC’s hard disk, and is loaded in the main memory of the PC, when the PC is booted, to be executed by the control of the processor 312.”), 4:51-67 (“FIG. 3 depicts, in block diagram form, the structure of the software instructions stored in the memory 322 (or a computer readable medium) of the PC 216 and executed by the processor 312 to provide functions for the handset or for the headset and for the display device 316. The messages transmitted among the software elements or modules 300, 320, 304, 324, and 308 and hardware elements 316 and 328 are conveyed via a standard operating system not illustrated in FIG. 3. Drivers 320 and 324 provide the standard software interface to sound card 212 and LAN card 220, respectively. Such drivers are well-known to those skilled in the art.”), Fig. 3. A POSITA would have understood that the types of storage medium disclosed by Becker, including the main memory of a PC, are forms of non-transitory computer readable storage medium, as opposed to transitory signals. Therefore, Becker teaches a non-transitory computer readable medium storing computer program code that, when performed by at least one processor, causes at least the operations as described for limitations [17b]-[17c] and claims 18-20 to be performed.

116. For the reasons set forth above, Grounds 3-4 teach this preamble.

[1b]/[11b]/[17b] causing a display of a device to display an object, relating to an application, when the device is in a locked mode; and

117. Becker teaches limitations [1b], [11b], and [17b]. First, as explained above, Becker teaches **a device** (e.g., PC). *Supra* §IV.B.2[1a]; Ex-1006, 3:54-64, 4:51-67, Figs. 2-3.

118. Becker teaches that the device has **a locked mode** (e.g., screen saver mode). Becker's screen saver mode, also referred to as "screen blanker" mode, is a locked mode—it is a "password-protected" mode of operation for a PC that is locked from normal use until the user unlocks the PC by entering a password. Ex-1006, 3:6-16 ("In another configuration, the graphical user interface is continuously visible even when the password-protected screen saver is engaged. As noted, no other software applications can be accessed while the password-protected screen saver is engaged."), 6:4-16 ("[T]he application 308 determines, based on the input, whether or not to terminate the screen saver display. When the input is defined information, the display is terminated, and the application repeats decision diamond 404 after a selected time. The defined information is typically a password but can include other information[.]"), 5:51-56 ("[W]hen the PC is in [screen saver] mode the PC user is unable to access a defined set of applications unless specified input, typically a password, is received from the user."), Fig. 4.

119. Becker's screen saver is a locked mode that allows limited interaction with a defined set of applications without fully activating them (*infra* §IV.B.2[1c]/[11c]/[17c]) but otherwise prohibits access to normal applications. Ex-1006, 3:20-29 ("It can allow for limited calling with specified call permissions (or limited calling to specified destinations) while the PC is password protected."), 2:61-3:3 ("No other software application may be accessed while the display is in the screen saver mode."), 5:51-56 ("In step 400, it is assumed that the PC 216 is activated and therefore is not currently in screen saver or blanker mode. As will be appreciated, when the PC is in this mode the PC user is unable to access a defined set of applications unless specified input, typically a password, is received from the user."), 3:12-16 ("[N]o other software applications can be accessed while the password-protected screen saver is engaged"), 4:25-40 ("[W]hen the device is in screen saving mode the ability of the user to access/enable other software applications, such as those stored in PC 216, is restricted."), 6:17-26, 6:48-60, 2:40-49. The '741 patent confirms that Becker's screen saver mode falls within the scope of the recited "locked mode."⁵

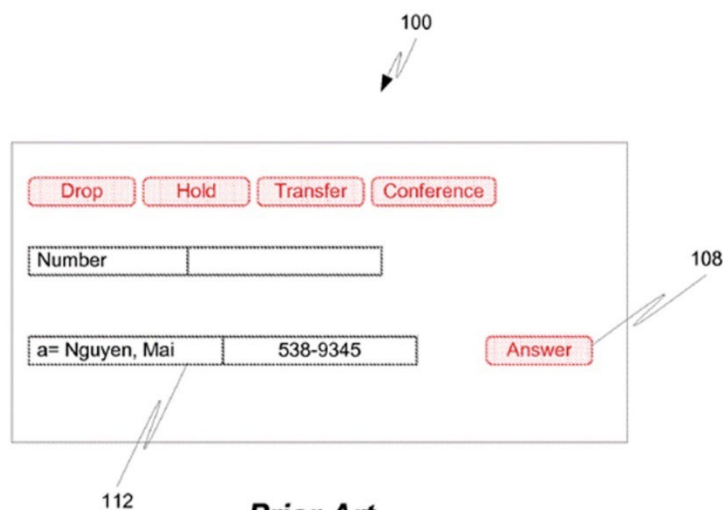
⁵ For example, the '741 patent discusses "screen saver" as locked mode. '741 patent, 4:17-19 ("In alternate embodiments, the icon 206 might move about the screen, in the *idle and locked mode, such as would a screen saver.*"). In addition, like Becker, the '741 patent's embodiments allow limited user interaction to permitted applications when in locked mode. '741 patent, 11:32-45 ("The disclosed embodiments generally provide for a user to select an application or set

120. Becker further teaches **causing a display of a device** (e.g., PC) to **display an object** (e.g., button in a GUI display), **relating to an application** (e.g., communication application), when the device is locked. For example, Becker discloses “a personal computer that includes[] a display device 316 for displaying image information.” Ex-1006, Abstract. Becker displays a GUI related to a communication application running on the PC. When the PC receives an incoming call during screen saver mode, the device changes the current display to include or be replaced by the GUI display of the communication application. Ex-1006, 5:1-24 (“When the display device 316 is in screen saving mode, the screen saver application 308, in response, changes the current display of the display device 316 to include or be replaced by a display including the softphone GUI.”), 6:17-26 (“When the input is not the defined information, the application 308, in decision diamond 416, determines if an incoming communication or contact notification has been received from the event notification agent 304. ... If so, the application, in step 422, displays the GUI for the communication application (e.g., softphone or instant messaging application)[.]”), 2:47-57, 2:61-3:11. Alternatively, Becker teaches that the GUI display “is always visible on the screen saver display.” Ex-

of user interface elements that a device will open to from an idle and locked mode.”), 1:41-53 (“In one embodiment the method includes providing a touch or proximity screen device in an idle mode. ... The term idle mode generally be referred to as a reduced active region that can be manipulated in limited ways, such as by the unlocking gesture or interaction described herein.”).

1006, 6:48-7:17. Either option satisfies the challenged claims because, in both instances, the PC displays the GUI window for the communications application, which includes button objects, while the PC is locked. Ex-1006, 5:1-24, 6:48-7:17.

121. Becker teaches that the GUI display includes buttons, e.g., “Answer,” “Drop,” “Hold,” “Transfer,” and “Conference” buttons. Ex-1006, 6:48-7:17 (“[T]he GUI display of the communications application, such as the display of FIG. 1A or 1C, (or an icon, such as that of FIG. 1B, associated with the display) is always visible on the screen saver display.”), Figs. 1A (color added):⁶



**Prior Art
Fig. 1A**

⁶ Becker applies its teachings to a prior art GUI application window. Ex-1006, 1:29-47, 3:40-42, 3:50, 4:25-40, 5:25-47, 6:4-16, 6:48-7:17, Figs. 1A-1D, Fig. 5. As Becker explains, application windows in GUIs were known in the art, and Becker utilizes a known GUI window to explain its additional features, e.g., allowing access to a limited set of functions when the PC is locked. Ex-1006, 5:1-24, 6:48-7:17, 1:29-47, 3:40-42, 3:50; *supra* §IV.B.1 (explaining how Becker’s prior art teachings are used to illustrate its embodiments and the motivation to combine Becker’s prior art teachings with its embodiments).

122. Each button, including the “Answer” button, is **an object, relating to an application** (e.g., communication application), that allows users to interact with the communication application. Ex-1006, 1:29-47 (“[T]he user may answer the call by selecting the softphone software application (such as by clicking on the corresponding icon 104 of FIG. 1B) and selecting a button 108 labeled ‘answer.’”), 6:48-7:17, Fig. 1A. Each button is a distinct object displayed on the graphical user interface of the application window for the user to click, which causes the application to perform the corresponding operation. *Id.*

123. Becker teaches displaying the application GUI display, including its “Answer” button, **when the device** (e.g., PC) **is in a locked mode** (e.g., screen saver mode). Ex-1006, 6:48-7:17 (“[T]he GUI display of the communications application, such as the display of FIG. 1A or 1C, (or an icon, such as that of FIG. 1B, associated with the display) is always visible on the screen saver display. The user may freely access all of a restricted set of the features of the communications application to accept incoming or place outgoing contacts or otherwise conduct communications; however, the user is prevented from accessing other applications on the PC.”), 5:1-24 (“*When the display device 316 is in screen saving mode*, the screen saver application 308, in response, changes the current display of the display device 316 to include or be replaced by a display including the softphone GUI. The user may perform normal softphone operations permitted by the selected

GUI display and its related children displays without the operations being blocked by the screen saver application.”), 6:17-26 (“[T]he application 308, in decision diamond 416, determines if an incoming communication or contact notification has been received from the event notification agent 304. ... If so, the application, in step 422, displays the GUI for the communication application (e.g., softphone or instant messaging application) (without receiving the defined information from a user) but continues to restrict the user’s access to a defined set of applications.”), 2:61-3:11 (“When the communication device is receiving the communication, a graphical user interface associated with the communication application is displayed to permit the user to answer the communication without first disabling or terminating the screen saver mode.”), 2:42-55 (“(a) [W]hen the display is in a screen saver mode, ... (c) when the event is a member of the defined set of events, displaying on the display (while the display is in the screen saver mode) image information associated with the event. The event can be any event generated by an application running on the personal computer, such as receipt of an incoming communication from a third party for a user of the personal computer, receipt of an outgoing communication from the user to the third party, and receipt of a reminder of a (previously) scheduled event.”), Figs. 1A-1C.

124. For the reasons set forth above, Becker teaches limitations [1b]/[11b]/[17b]. Grounds 3-4 therefore teach limitations [1b]/[11b]/[17b].

[1c]/[11c]/[17c] enabling a user to select the object while the device is in the locked mode, in order to determine a state, function or utility of the application without fully activating the application while the device remains in the locked mode.

125. Becker teaches limitations [1c], [11c] and [17c]. First, Becker teaches **enabling a user to select the object** (e.g., button in the GUI display) **while the device is in the locked mode** (e.g., screen saver mode). As explained above, Becker teaches that when the PC is in the screen saver mode, an “Answer” button related to the communication application is displayed, in a GUI display, to “permit the user to answer the communication” without disabling or terminating the screen saver mode. Ex-1006, 2:61-3:11, 6:48-7:17, 5:1-24, 6:17-26, 2:47-57, Fig. 1A; *supra* §IV.B.2[1b]/[11b]/[17b]. Becker teaches that users select the GUI display, including buttons (e.g., “Answer,” “Drop,” “Hold,” “Transfer,” and “Conference” buttons), to perform permitted operations when the PC is in the screen saver mode. Ex-1006, 5:11-24 (“When the display device 316 is in screen saving mode, the screen saver application 308, in response, changes the current display of the display device 316 to include or be replaced by a display including the softphone GUI. The user may perform normal softphone operations permitted by the GUI display without the operations being blocked by the screen saver application. For example, for voice and instant messaging applications the user may answer the communication, drop the communication, place the communication on hold,

transfer the communication, conference in other parties, hangup or terminate the communication, send and receive voice or nonvoice communications, view the communications, and the like.”), 4:25-40, 6:48-7:17, Figs. 1A-1C. In short, Becker enables a user to select the buttons on the application GUI display, while the device remains in the screen saver mode, to “answer,” “drop,” “hold,” “transfer,” or “conference” an incoming call. *Id.*

126. Becker teaches users answering and terminating calls by “selecting” the corresponding buttons, e.g., by clicking on them. Ex-1006, 1:29-1:47 (“[T]he user may answer the call by *selecting the softphone software application (such as by clicking on the corresponding icon 104 of FIG. 1B) and selecting a button 108 labeled ‘answer.’* ... When the call is answered, the display changes automatically to the display 120 of FIG. 1D, which includes buttons for drop 124, hold 128, transfer 132, conference 136, and hangup 140. *The call is terminated by the called user by selecting the hangup button 140.*”), 1:61-66 (“When the user hears the IP softphone ringing, the user must unlock the screen blanker (typically by inputting a selected set or sequence of characters including or followed by a password) to access the PC, select the software application, and activate the soft button labeled ‘Answer.’”), Figs. 1A-1D. Becker therefore teaches enabling a user to select buttons on the application GUI. Additionally, this would have been obvious because buttons in a GUI were displayed specifically for the purpose of allowing

users to click and thereby select them, as Becker teaches. *See* Ex-1006, 1:29-1:47, 1:61-66, Figs. 1A-1D; *e.g.*, Ex-1005, ¶26 (“Each of the application elements may be selectable items which may further have as shown here one or more subordinate elements or features, generally identified as elements 28, extended horizontally across the display”), ¶29 (“Note, the example of FIG. 2A provides some additional information as for example when a particular icon or element 29 is selected.”), Figs. 2-3, ¶35 (“Also shown may be one or more (*e.g.*, a list) of selections or selectable items, here *e.g.*, MP3 files 32, 33, 34 and/or 35, *inter alia* (including those shown and/or unshown in FIG. 3), which may be played with/on/by the software application.”), ¶36 (“In this first example of a means for implementation of the present invention, a generally highlighted area 36 is shown which provides, only in direct relationship to a specific item, here item or MP3 file 32, an expanded display of a multifunctional set 38 of features, here, operational icons, *e.g.*, icons 38 *a*, 38 *b*, 38 *c* and 38 *d*. These operational icons 38 in being operational are thus selectable as well.”), ¶44 (“Toolbox items can be selected by pressing select key, see the gleaming phone icon 48 of FIG. 5C[.]”), ¶49 (“As shown by the gleaming icon 73 in FIG. 9B, the user selects a function (here, an exemplar save-function).”).

127. Becker further teaches enabling a user to select the object (*e.g.*, button) while the device (*e.g.*, PC) is in the locked mode (*e.g.*, screen saver mode)

in order to determine a state, function or utility of the application. Becker enables a user to select buttons, including the “Answer” button, to determine a function of the application (e.g., the ability to answer, drop, hold, transfer, or conference incoming calls) during screen saver mode. These are functions of Becker’s communication application. Ex-1006, 1:29-47, 5:1-24, 2:61-3:11, 6:48-7:17. For example, Becker teaches that users select the “Answer” button in order to determine the function of the application, e.g., whether or not the communication application supports the functionality of answering incoming calls during screen saver mode. *E.g.*, Ex-1006, 2:61-3:11, 6:48-7:17. Likewise, Becker enables users to select the drop, hold, transfer, and conference buttons to determine whether or not the application supports that functionality during screen saver mode. *Id.*

128. Becker also enables a user to select buttons to determine a utility of the application (e.g., whether or not the application can be used to answer, drop, hold, transfer, or conference incoming calls) during screen saver mode. Becker’s communication application is utilized to answer, drop, hold, transfer, or conference incoming calls, and Becker enables users to determine whether or not those uses are supported during screen saver mode by clicking on the corresponding button. *E.g.*, Ex-1006, 1:29-47, 5:1-24, 2:61-3:11, 6:48-7:17.

129. Patent Owner’s district court representations regarding how the claims can be satisfied confirm that Becker’s teachings fall within the scope of this

limitation. For example, Patent Owner broadly alleges that this limitation is satisfied by icons in the Android UI where the user can interact with the icon during the alleged locked mode, including a range of objects for changing Wi-Fi and Bluetooth operation. Ex-1009, 000007-10 (accusing possible user interactions for “pressing or tapping an object, such as the object related to a wi-fi application or Bluetooth application, determines the state, function, or utility of that related application without fully activating the application.”).

130. Becker further teaches enabling a user to select the object (e.g., button) in order to determine a state, function or utility of the application **without fully activating the application** while the device remains in the locked mode. Becker teaches that the communication application is not fully activated because the user may access only a restricted set of the features that only “allow[s] for limited calling with specified call permissions (or limited calling to specified destinations) while the PC is password protected.” Ex-1006, 3:20-29. Becker further teaches limiting user access to “a restricted set of the features of the communications application” while the PC is in the screen saver mode, where, for example, the PC “permit[s] only incoming communications to be answered but the user restricted from making or initiating outgoing communications.” Ex-1006, 6:48-60:

The user may freely access all of a restricted set of the features of the communications application to accept incoming or place outgoing contacts or otherwise conduct communications; ... For example, the incoming GUI display, such as that of FIG. 1A, can be displayed to permit only incoming communications to be answered but the user restricted from making or initiating outgoing communications. ... To prevent unauthorized use of the communications application to make outgoing communications, the communications application may be enabled for a specified or defined set of telephone numbers or electronic addresses unless the screen saver is terminated.

Therefore, when in locked mode, Becker's communication application is not fully activated and is only enabled for limited operations with a limited set of the features. *See id.*

131. As Becker explains, this allows the user to temporarily answer calls while the PC remains locked, without fully activating the communications application. When a user selects buttons, e.g., selecting the "Answer" button to answer a call, the communication application is only temporarily activated, where the display of the device would return to the screen saver display after the call. Ex-1006, 5:25-47 ("When the communication is terminated, ... the softphone application 300 transmits a disconnect message to PBX 204 to cause the communication to be terminated and event notification agent 304 to cause the display of the display device 316 to be returned to the original screen saver

display.”); 6:27-44 (“In step 428, the application 308 removes the softphone GUI display from or otherwise clears the screen of the display device 316 and redispays the screen saver display.”), Fig. 4:

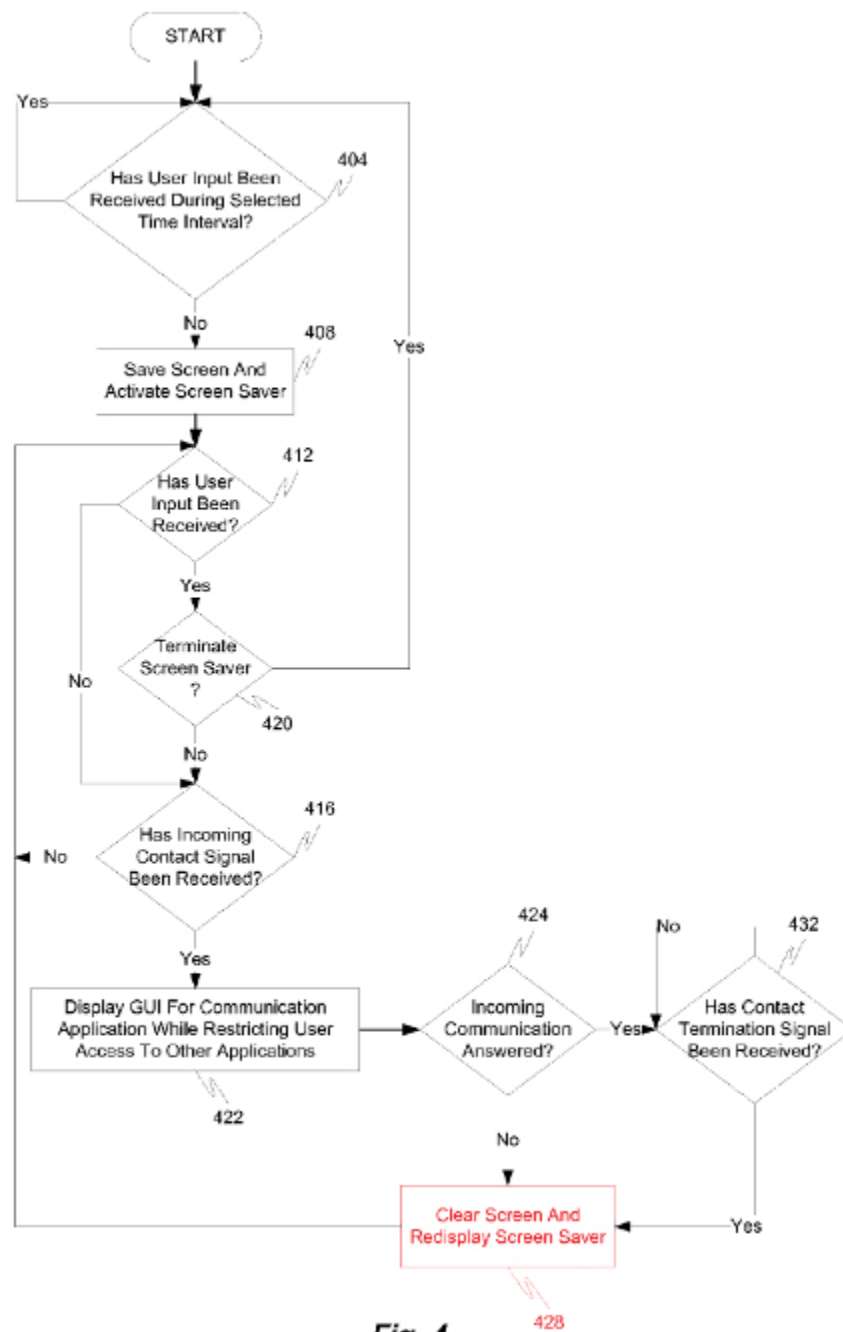


Fig. 4

132. Since Becker's communication application is only temporarily activated with limited functionality, it serves Becker's underlying purpose of allowing users to temporarily use the application—e.g., to answer calls—while otherwise “prevent[ing] unauthorized use of the communications application[.]” Ex-1006, 6:63-7:3. Becker explains that allowing the communication application to be only temporarily activated helps prevent unauthorized users from “freely access[ing] information on the PC ... at the user's expense, via the communication application's GUI.” Ex-1006, 5:42-47. Becker therefore enables a user to select the object (e.g., button) in order to determine a state, function or utility of the application without fully activating the application while the device remains in the locked mode.

133. For the reasons set forth above, Becker teaches limitations [1c]/[11c]/[17c]. Grounds 3-4 therefore teach limitations [1c]/[11c]/[17c].

3. Dependent Claims 2 and 13

- 2. The apparatus as claimed in claim 1, wherein the computer program code is configured, working with the at least one processor, to cause at least the following to be performed:**
- 13. The method as claimed in claim 11, further comprising:**
responding to user selection of the object by illuminating or highlighting the object.

134. Ground 4 teaches claims 2 and 13. Becker teaches the apparatus of claim 1, including computer program code that is configured, working with the at

least one processor, to cause the steps described in this Declaration to be performed, and the method of claim 11. *Supra* §IV.B.2. The combination of Becker with Nirhamo (Ground 4) further teaches the additional limitations of claims 2 and 13.

135. As explained above, Becker teaches the user selecting an object related to an application. *Supra* §§IV.B.2[1b]-[1c]. Nirhamo further teaches **responding to user selection of the object by illuminating or highlighting the object**. Nirhamo “relates to improved presentation, navigation, selection and/or operation options for portable communication devices, particularly, user interface options involved on the display screens thereof.” Ex-1005, ¶1. Nirhamo discloses device operations, including highlighting items on the display when they are selected. Ex-1005, Abstract, ¶4, ¶¶29-31, ¶¶36-39, ¶41, ¶51, ¶71. For example, Nirhamo teaches that “when a particular icon or element[] is selected[,] [t]his element[] may then be highlighted in some way (as here by a distinctively colorized box framed therearound, although other means are also usable herewith).” Ex-1005, ¶29, ¶51 (“Generally, focused-upon items are shown here marked with dotted backgrounds (though these could be highlighted otherwise, e.g., by being brightened or gleamed relative to other selection alternatives or by being presented in a distinctive colour, or other style, inter alia). Up and down keys of a joystick (or other cursor movement implementation such as a four or eight

way button) can be used to select the establishment of a focus or highlighting on an item, e.g., ‘Item 1’ element 81, ‘Item 2’ element 82, and/or ‘Item 3’ element 83. ... In these embodiments, the ‘items’ 81, 82, 83 may be considered as either selectable items or features as these terms are used throughout.”).

136. A POSITA would have been motivated to apply Nirhamo’s user-selection teachings with Becker because Becker teaches the user selecting buttons (e.g., “Answer” button) to determine whether or not the associated operation is permitted (e.g., whether or not the user is able to answer incoming calls during screen saver mode). *Supra* §IV.B.[1c]. It was customary for device interfaces to highlight the user’s selection so that the user could see and confirm what was selected. For example, Becker teaches the user selecting buttons to perform operations. Ex-1006, 5:11-24, 4:25-40, 6:48-7:17, Figs. 1A-1C; *supra* §IV.B.2[1c]. For this to happen, it would have been obvious to highlight the selected icon, so that the user can see which application or operation was currently selected (e.g., for forwarding, replying to, or opening a message). This was a common, universal feature in device interfaces (e.g., PC interface) since the early GUIs of the 1990s. If the user’s selection were not highlighted in some fashion, it would have been difficult for the user to navigate the computer interface, as there would not be a clear way for the user to discern whether the correct object had been selected.

137. Highlighting the user's selection would not have changed the principle of operation of either reference. Indeed, it would have furthered Becker's teachings by facilitating the user's selection of icons. A POSITA would have had a reasonable expectation of success in combining Becker with Nirhamo's teachings for highlighting user selections because the combination would have been a common, routine, and expected operation that is suggested by Becker's teaching of user selection. Highlighting user selection was implied by Becker because that is how device interfaces (e.g., computer interfaces) had operated for many years. Ex-1011, 000023-57, Fig. 8-1.

138. The combination of Becker with Nirhamo therefore teaches responding to user selection of the object by illuminating or highlighting the object. Ground 4 thus teaches claims 2 and 13.

4. Dependent Claims 3 and 14

- 3. The apparatus as claimed in claim 2,**
- 14. The method as claimed in claim 11,**
- wherein a movable object is provided to enable the user to select the object.**

139. Ground 4 teaches claims 3 and 14. The combination teaches the apparatus of claim 2, and the method of claim 11. *Supra* §§IV.B.2, V.B.3. The combination of Becker with Nirhamo (Ground 4) further teaches the additional limitations of claims 2 and 13.

140. The combination teaches that a movable object (e.g., cursor) is provided to enable the user to select the object. As explained above, Becker teaches that a user selects an object relating to an application. *Supra* §IV.B.2. Nirhamo teaches using movable objects such as a cursor for user operations (e.g., selecting items in the display). Ex-1005, ¶39 (“[T]he user can move an emphasized or otherwise highlighted cursor or visual selection representation (here shown by bolding and/or the darker coloring of the play/pause button 38 a) to select the desired functional operation to be performed for the selected item 32[.]”), ¶51 (“Up and down keys of a joystick (or other cursor movement implementation such as a four or eight way button) can be used to select the establishment of a focus or highlighting on an item, e.g., ‘Item 1’ element 81, ‘Item 2’ element 82, and/or ‘Item 3’ element 83.”), ¶23 (“[T]he keypad 2 has a first group 7 of data entry buttons or keys as alphanumeric keys, two softkeys 8, and a scroll-key 10 (up/down and/or right/left and/or any combination thereof) for moving a cursor in the display 3. An alternative hereto may be a four-way button, an eight-way button or a joystick, track ball, roller or other cursor controller [.]”), ¶38 (“The operator or user of the phone can then initiate or otherwise change the desired function to be used directly in the highlighted area 36 using phone cursor control keys, such as for example, an arrow key or keys, see multidirectional key 10 in FIG. 1 (alternative multidirectional keys, joysticks, rollers etc. or

individual right and left or up and down keys may otherwise be used as well).”). A cursor is a movable object because it is an object moved based on user input, to manipulate its position so that the user can navigate through the user interface and perform operations such as selecting items on the display.

141. A POSITA would have been motivated to combine Nirhamo’s cursor teachings with Becker because it provided additional context for the known user interface taught by Becker. Specifically, Nirhamo describes additional details of pointing, clicking, selecting, etc., which Becker already suggests because Becker teaches conventional PC operations such as selecting items on the display. Using input/point devices (e.g., a cursor) to facilitate user interactions has been known for decades and was widely applied across different operating systems and software applications. *See* Ex-1007, ¶29 (“[S]ome of the user interfaces and associated processes may be applied to other devices, such as personal computers and laptop computers, which may include one or more other physical user-interface devices, such as a click wheel, a physical keyboard, a mouse and/or a joystick.”), ¶40 (“[I]nput controller(s) 160 may be coupled to any (or none) of the following: a keyboard, infrared port, USB port, and a pointer device Such as a mouse.”); Ex-1010, 3:22-43, 8:13-26, 16:32-47 (“In a simple graphics model, object may be selected by clicking directly on the object with the mouse or dragging a selection box across several objects using a mouse tracker.”), 37:51-38:3 (“A graphical user

interface is manipulated by moving a mouse, clicking on objects to select them, dragging objects to move or copy then, and double-clicking to open them.”), 41:8-20 (“Users of an object oriented operating system’s graphical user interface often move a mouse, click on objects to select them, drag objects to move or copy then, and double-click to open an object.”), Fig. 1A.

142. Therefore, the combination teaches providing a movable object (e.g., cursor) in the user interface to enable the user to select the object. Ground 4 thus teaches claims 3 and 14.

5. Dependent Claim 4

4. The apparatus as claimed in claim 2, wherein the object is highlighted or illuminated to a lesser extent than that associated with a standard mode setting.

143. Ground 4 teaches claim 4. The combination teaches the apparatus of claim 2. *Supra* §IV.B.3. The combination of Becker with Nirhamo (Ground 4) further teaches the additional limitations of claim 4.

144. As explained above, the combination teaches responding to user selection of the object by illuminating or highlighting the object. *Supra* §IV.B.3. The combination further teaches that **the object is highlighted or illuminated to a lesser extent than that associated with a standard mode setting**. As an initial matter, Nirhamo teaches illuminating or highlighting an item by changing its brightness. Ex-1005, ¶51 (“Generally, focused-upon items are shown here marked

with dotted backgrounds (though these could be highlighted otherwise, e.g., by being brightened or gleamed relative to other selection alternatives or by being presented in a distinctive colour, or other style, inter alia).”), ¶43 (“In such a single-item listing, the display 40 including a list 41 as shown first in FIG. 5A, a focus 43 (by gleaming, color or brightness or other highlight change) is placed on an item 42 in the list 41.”), ¶39 (“Thus, in this example, the user can move an emphasized or otherwise highlighted cursor or visual selection representation (here shown by bolding and/or the darker coloring of the play/pause button 38 a) to select the desired functional operation to be performed for the selected item 32[.]”).

145. Nirhamo further discloses dimming items that are unavailable or otherwise incompatible with an operation. Ex-1005, ¶36, ¶¶53-54 (“... Here only the currently focused item is de-emphasized or dimmed when the action indication is focused on ‘Cancel.’ See ‘Item 2’ element number 82 b. This de-emphasis shows the direct relationship of the action ‘Cancel’ to the highlighted item 82 b, specifically, that the action is not applicable to the item.”), ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C. These items have “de-emphasized” representations that contrast with available items to indicate their unavailability. Ex-1005, ¶¶53-54, ¶36 (“Note, the other selectable items, e.g. items 33, 34, 35, inter alia, have contrasting

non-highlighted representations.”). Dimming refers to a darker, i.e., reduced brightness level, meaning a dimmed object is illuminated to a lesser extent.

146. Becker teaches that some buttons are unavailable when the device is in screen saver mode because the associated operations are not permitted during the locked state. *Supra* §IV.B.2[1c]; Ex-1006, 3:20-29, 6:48-60. A POSITA would have been motivated to apply Nirhamo’s teachings by dimming these unavailable buttons because it indicates to the user that the operation is not permitted during screen saver mode. *See* Ex-1005, ¶36, ¶¶53-57, Figs. 10B-10C, Figs. 11A-11. This would have provided a simple, intuitive way to implement Becker’s teachings and indicate to the user which icons were not permitted, consistent with general device interfaces (e.g., computer interfaces). A POSITA would have had a reasonable expectation of success in combining these teachings because it was simply a matter of dimming a button on the computer display, and reducing brightness was a common and simple setting in computer devices by 2007.

147. Therefore, Becker teaches that some buttons are unavailable during the screen saver mode (*supra* §IV.B.2), and the combination with Nirhamo teaches dimming those buttons, meaning they are highlighted or illuminated to a lesser extent than a standard mode of operation, when all applications and operations are available. Ex-1005, ¶36, ¶¶53-54, ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C. In other words, when the device is in a standard, unlocked mode of operation, those

operations are available and no longer dimmed. When the user selects those buttons to determine whether or not the associated operations are permitted, Nirhamo teaches highlighting items to indicate the user selection, for example with “a distinctively colorized box framed” as explained for claim 2 (Ex-1005, ¶29, ¶51; *supra* §IV.B.3), and also dimming and de-emphasizing the item to indicate its unavailability (Ex-1005, ¶36, ¶¶53-54, ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C).

148. Ground 4 therefore teaches claim 4.

6. Dependent Claims 5 and 15

5. The apparatus as claimed in claim 1,
15. The method as claimed in claim 11,
wherein the object represents an application that is present or available in the device.

149. Grounds 3-4 teaches claims 5 and 15. Becker teaches the apparatus of claim 1, and the method of claim 11. *Supra* §IV.B.2. Becker further teaches the additional limitations of claims 5 and 15.

150. As explained above, Becker teaches causing a display of a device to display an object relating to an application that a user selects the object to determine a state, function or utility of the application.

Supra §§IV.B.2[1b]/[11b]/[17b]-[1c]/[11c]/[17c]. Becker further teaches **the object represents an application that is present or available in the device.** For

example, Becker teaches that an application (e.g., communication application) is present or available in the PC so that a user may answer incoming calls using the application by selecting the object relating to the application (e.g., button in the GUI display). *E.g.*, Ex-1006, 2:40-3:11, 5:1-47, 6:48-7:17. The object therefore represents an application that is present or available in the device. Grounds 3-4 therefore teach claims 5 and 15.

7. Dependent Claim 6

6. The apparatus as claimed in claim 1, wherein the computer program code is configured, working with the at least one processor, to cause at least the following to be performed: causing the display to display another object, relating to another application, when the device is in a locked mode.

151. Grounds 3-4 teach claim 6. Becker teaches the apparatus of claim 1.

Supra §IV.B.2. Becker further teaches the additional limitations of claim 6.

152. Becker teaches **causing the display to display another object, relating to another application, when the device is in a locked mode**. For example, Becker teaches displaying another object (e.g., icon) for another application (e.g., email application) during screen saver mode. Ex-1006, 7:18-30:

[T]he event notification agent 304 monitors for events other than the arrival of an incoming communication. The events would be generated by an application running on the PC. For example, calendaring or scheduling software, such as Microsoft Outlook™ or an email application, can generate reminders that are to be displayed to the PC

user. Normally, the reminders are blocked by the screen saver. When the event notification agent detects the occurrence of such an event 304, the agent would cause *the screen saver display to be altered to include the display associated with the event or an icon associated therewith*. The altered display could be replaced automatically by the full screen saver display after a selected period of time.

153. Grounds 3-4 therefore teach claim 6.

8. Dependent Claim 7

7. The apparatus as claimed in claim 6, wherein the another object represents an application that is present or available in the device.

154. Grounds 3-4 teach claim 7. Becker teaches the apparatus of claim 6.

Supra §IV.B.7. Becker further teaches the additional limitations of claim 7.

155. As explained above, Becker teaches causing the display to display another object relating to another application, when the device is in a locked mode (e.g., screen saver mode). *Supra* §IV.B.7. Becker further teaches that the another object represents an application (e.g., Microsoft Outlook or an email application) that is present or available in the device. Ex-1006, 7:18-30. Grounds 1-2 therefore teach claim 7.

9. Dependent Claim 8

8. **The apparatus as claimed in claim 6, wherein the another object indicates that the another application is not activatable when the device is in the locked mode.**

156. Ground 4 teaches claim 8. Becker teaches the apparatus of claim 6.

Supra §IV.B.7. Becker further teaches the additional limitations of claim 8.

157. The combination of Becker and Nirhamo teaches that **the another application** (e.g., Microsoft Outlook or an email application) **is not activatable when the device** (e.g., PC) **is in the locked mode** (e.g., screen saver mode). As explained above, the combination teaches that an object in the display (e.g., dimmed icon) indicates that the object is unavailable or otherwise incompatible with an intended action/operation. *Supra* §IV.B.5; Ex-1005, ¶53 (“Items in the list 86, see particularly items 81 *a*, 82 *a*, and 83 *a*, are de-emphasized or dimmed in FIG. 10B, as indicated here by the distinctive less bold font, so that the user knows *they are not part of the possible or intended “Cancel” action suggested here by the action focus/indication on the “Cancel” action element 85 a.*”), ¶54, ¶¶56-57 (“See for example, the display 90 of FIG. 11A which shows three items 91, 92 and 93, where however, the third such item 93, ‘item 3,’ is shown dimmed (indicated by the distinctive, less-bold font). This is dimmed when the first action in this example, here ‘Act 1,’ element 94, is highlighted (see the dotted background thereof), thereby *indicating that item 93 is unavailable for*

or otherwise incompatible with operation by 'Act 1' 94. ... FIG. 11B shows in a display 90 a an alternative situation when for example the 'Act2' element 95 a has been selected and then corresponding unavailable items 91 a and 92 a ('items 1 and 2[']) are dimmed.'), ¶60, Figs. 10B-10C, Figs. 11A-11C.

158. Becker teaches that when the PC is in screen saver mode, the user is “prevented from accessing other applications on the PC.” Ex-1006, 6:48-63 (“The user may freely access all of a restricted set of the features of the communications application to accept incoming or place outgoing contacts or otherwise conduct communications; however, the user is prevented from accessing other applications on the PC.”). Becker therefore teaches applications (e.g., Microsoft Outlook or an email application) other than permissible applications (e.g., communication application) are not activatable when the PC is in the screen saver mode.

Therefore, based on the combined teachings, a dimmed icon indicates that the another application (e.g., Microsoft Outlook or an email application) is not activatable when the device is in the locked mode. Ground 4 therefore teaches claim 8.

10. Dependent Claims 9 and 16

- 9. The apparatus as claimed in claim 1,**
- 16. The method as claimed in claim 11,**
- wherein the display is a touch screen display, and enabling a user to select the object comprises enabling the user to select the object using the touch screen display.**

159. Ground 4 teaches claims 9 and 16. Becker teaches the apparatus of claim 1, and the method of claim 11. *Supra* §IV.B.2. The combination of Becker with Nirhamo (Ground 4) further teaches the additional limitations of claims 9 and 16. For example, Nirhamo teaches that touch screen could be used to implement the operations it discloses. Ex-1005, ¶23 (“Touch screen functionality could also be used.”). Moreover, using touch screen displays to facilitate user interactions (e.g., selecting objects displayed on the touch screen) was widely known in the art well before the timeframe of the ’741 patent. Ex-1007, Abstract, ¶¶7-11, ¶77 (“The touchscreen may display one or more graphics. ... [A] user may select one or more of the graphics by making contact or touching the graphics, for example, with one or more fingers 202 (not drawn to scale in the figure) or a stylus (not shown in the figure).”), ¶119, ¶125 (“In some embodiments, the user interface includes a finish icon 510 having text such as ‘done,’ ‘okay,’ or ‘save.’ A user selection of this icon causes the portable device to complete the user configuration process and return to the user interface[.]”), ¶126, Fig. 2; Ex-1010, 3:22-43.

160. It would have been obvious to modify Becker based on Nirhamo's teachings such that the display of the device is a touch screen display. A POSITA would have understood that devices with touch screen displays allow users to select objects using the touch screen.

161. A POSITA would have been motivated to combine Nirhamo's touchscreen teachings to Becker for several reasons. Becker teaches an application running on a computer. Ex-1006, 3:54-4:24. Touchscreens were common features on laptop computers and mobile devices by 2007, and Nirhamo teaches a device with a touchscreen. Ex-1005, ¶23. Therefore, the inclusion of a touchscreen with Becker's teachings would have used a known technique to improve similar devices in the same way.

162. Moreover, the combination would have had several advantages. For example, touchscreens offered an intuitive user interface because users could simply touch the screen to interact with the computer directly. For that reason, touchscreens were often touted as a marketing feature in laptops. In addition, touchscreens allowed users to quickly navigate through applications with simple operations such as tapping and swiping, which speeds up user operations and improves operation efficiency. Therefore, a POSITA would have had ample motivation to combine Nirhamo's teachings with Becker.

163. The combination would not have changed the principle of operation for Becker or Nirhamo. The combination simply adds a touchscreen to Becker's computer, which was a common computer peripheral that adds an additional way for the user to interact with the computer but does not alter the manner in which Becker's teachings were applied to a computer. Nor would it have changed Nirhamo's principle of operation since the touchscreen still would have operated as a touchscreen. A POSITA would have had a reasonable expectation of success in combining these teachings because it was simply a matter of replacing the display with a touch screen display, and touch screen displays were common and widely used by 2007.

164. Grounds 3-4 therefore teach claims 9 and 16.

11. Dependent Claim 10

10. The apparatus as claimed in claim 1, wherein the apparatus is the device.
--

165. Grounds 3-4 teach claim 10. Becker teaches the apparatus of claim 1. *Supra* §IV.B.2. As explained above, Becker teaches its PC as an apparatus that comprises the processor and memory that cause the operations described in limitations [1b]/[11b]/[17b]-[1c]/[11c]/[17c] to be performed. *Supra* §IV.B.2. Becker therefore teaches that the apparatus is the device, i.e., the PC. Grounds 3-4 therefore teach claim 10.

12. Dependent Claims 12 and 18

12. The method as claimed in claim 11,
18. The non-transitory computer readable medium as claimed in claim 17,
wherein the object is a dimmed icon.

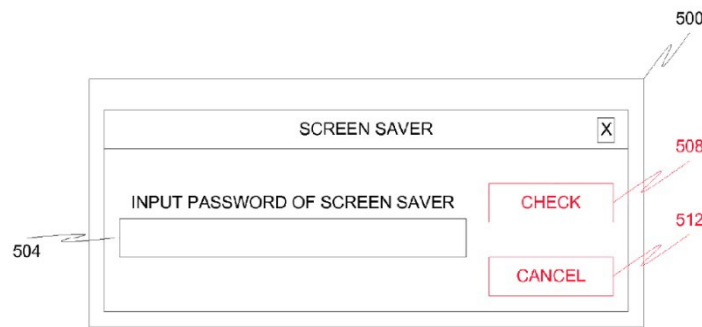
166. Ground 4 teaches claims 12, and 18. Becker teaches the method of claim 11, and the non-transitory computer readable medium of claim 17. *Supra* §IV.B.2. The combination of Becker with Nirhamo (Ground 4) teaches that **the object is a dimmed icon**. As explained above, Becker teaches that some buttons are unavailable during the locked mode because the associated operations are not permitted (*supra* §IV.B.2), and the combination with Nirhamo teaches dimming icons when the associated applications or operations are unavailable. *Supra* §IV.B.5; Ex-1005, ¶¶36, ¶¶53-54, ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C. The combination therefore teaches that the object is a dimmed icon, and thus teaches claims 12 and 18.

13. Dependent Claim 19

19. The apparatus as claimed in claim 17, wherein the object is an icon.

167. Grounds 3-4 teach claim 19. Becker teaches the apparatus of claim 17. *Supra* §IV.B.2. As explained above, Becker teaches that the object (e.g., button in a GUI) related to a communication application. *Supra* §IV.B.2; e.g., Ex-1006, 6:48-7:17, Figs. 1A-1D, Fig. 5. Becker teaches that the button in a GUI is an icon.

Ex-1006, 6:4-16 (“The display 500 of FIG. 5 provides the user with a password field 504 for entry of the password and optional *check and cancel icons 508 and 512.*”), Fig. 5 (color added):



Prior Art
Fig. 5

168. A POSITA would have understood this because according to Becker’s teachings, a button is a graphical representation of the associated application and its features. In addition, the use of icons that can be selected to cause an action was exceedingly well known. As mentioned above, the 1992 Macintosh Human Interface Guidelines has an entire 33 page chapter on icons. Ex-1011, 000025-57. It explains that “[i]cons are graphic representations of objects such as documents, storage media, folders, applications, and the Trash. Icons look like their real-world counterparts wherever possible. People can select, open, move, copy, and throw away icons. Figure 8-1 shows some icons displayed in the Finder.” Ex-1011, 000026. This is also confirmed by Nirhamo. Ex-1005, ¶39 (“In the particular example of FIG. 3 which shows involvement with a music or MP3 player

application, the functional icons, here exemplified by icons 38 *a*-38 *d*, may represent a music play/pause button 38 *a*, fast forward button 38 *b* (rewind shown but not separately identified), and/or sound level control 38 *c* (softer) and 38 *d* (louder).”), Figs .3A-3B.

169. Becker therefore teaches that the object is an icon. Grounds 3-4 thus teach claim 19.

14. Dependent Claim 20

20. The apparatus as claimed in claim 19, wherein the icon is a dimmed icon.

170. Ground 4 teaches claim 20. As explained above, Becker teaches the apparatus as claimed in claim 19 and that the object is an icon. *Supra* §IV.B.13. As explained above, the combination of Becker with Nirhamo (Ground 4) teaches that the objects include icons. *Supra* §IV.B.13. The combination further teaches dimmed objects. *Supra* §§IV.B.5, V.B.12. The combination therefore teaches that the icon is a dimmed icon. Ground 4 thus teaches claim 20.

V. BACKGROUND AND QUALIFICATIONS

171. This section contains a summary of my educational background, career history, publications, and other relevant qualifications. My full curriculum vitae is attached as Appendix 1 to this declaration.

172. I am an expert in the field of user interfaces and have been an expert in that field since before the relevant timeframe, and before the challenged patent

was filed. A copy of my *curriculum vitae* is provided as Appendix 1 to this report and provides a comprehensive description of my relevant experience, including my academic and employment history, publications, professional activities, and speaking engagements.

173. I received a Bachelor of Science degree in Computer Science with a minor in Electrical Engineering from Rensselaer Polytechnic Institute (“RPI”) in 1986. I received a Master of Science degree and a Ph.D. in Computer Science from New York University (“NYU”) in 1989 and 1992, respectively.

174. Since 1998, I have been a Professor of Computer Science at the University of Maryland, where I have had joint appointments at the Institute for Advanced Computer Studies and the College of Information Studies (Maryland’s “iSchool”) and am currently Professor Emeritus. I was also Associate Provost of Learning Initiatives and Executive Director of the Teaching and Learning Transformation Center from 2014 to 2018. I am a member and previous director of the Human-Computer Interaction Lab (“HCIL”), the oldest and one of the best known Human-Computer Interaction research groups in the country.

175. I am co-founder of the J.S. Bryant School, a therapeutic high school which we plan to open in 2025. I am also co-founder and co-director of the International Children’s Digital Library (“ICDL”), a web site launched in 2002 that provides the world’s largest collection of freely available online children’s books

from around the world with an interface aimed to make it easy for children and adults to search and read children's books online. I was also co-founder and Chief Scientist of Zumobi, Inc. from 2006 to 2014, a Seattle-based startup that is a publisher of content applications and advertising platforms for smartphones. I was also co-founder and prior Chief Technology Officer of Hazel Analytics, a data analytics company whose product sends alerts in warranted circumstances. In addition, I have for more than 25 years consulted for numerous companies in the area of user interfaces, including Logitech, Microsoft, the Palo Alto Research Center, Sony, Lockheed Martin, and NASA Goddard Space Flight Center.

176. For more than 30 years, I have studied, designed, and worked in the field of computer science and human-computer interaction. My experience includes 30 years of teaching and research, with research interests in human-computer interaction and the software and technology underlying today's interactive computing systems. This includes the design and implementation of user interfaces and software applications using a wide range of technologies including web servers and websites, databases, embedded systems, and mobile devices, including smart phones and PDAs.

177. At UMD, my research has been in the area of Human-Computer Interaction ("HCI"), a field that relates to the development and understanding of computing systems to serve users' needs. Researchers in this field are focused on

making universally usable, useful, efficient, and appealing systems to support people in their wide range of activities. My approach is to balance the development of innovative technology that serves people's practical needs. Example systems following this approach that I have built include Pad++ (1994 software for zoomable user interfaces), KidPad (1997 software to support children collaboratively writing stories), PadPrints (1998 software to graphically show interactive web browsing histories), PhotoMesa (2001 software for end users to browse personal photos), Pocket PhotoMesa (2004 software for browsing of personal photos on a mobile device), DateLens (2002 software for end users to use their mobile devices to efficiently access their calendar information), LaunchTile (2005 "home screen" software for mobile devices to allow users to navigate apps in a zoomable environment), SpaceTree (2001 software for end users to efficiently browse very large hierarchies), ICDL (as described above), and StoryKit (a 2009 iPhone app for children to create stories).

178. In January 2000, I personally bought and started using a Kodak DC280 digital camera, and at that point started collaborating with my colleague Ben Shneiderman on the PhotoFinder project. I joined group meetings, participated in group emails, and was an active participant in the project as a whole, using and testing PhotoFinder and giving feedback about its development throughout the year 2000. For example, in September 2000, I provided feedback about the details of

the interaction, including how annotation should be performed, how annotations should be displayed, and numerous details about the PhotoFinder interface as a whole.

179. After being involved with the PhotoFinder project, I decided to develop my own project that was initially focused on the visual presentation and interaction with photos. I started this project in January 2001 and called it PhotoMesa. I wrote a 2001 paper⁷ that describes PhotoMesa, a zoomable image browser using a novel 2D layout algorithm I called “quantum treemaps” to position the images on the screen in a way that attempted to fill the screen while keeping groups of related images together. As indicated in the figure below, PhotoMesa could display many hundreds of images at a time by showing small thumbnails, by showing larger thumbnails when the mouse hovered over a small thumbnail, and by allowing a user to zoom in to see a high-resolution version of the images:

⁷ Benjamin B. Bederson, *PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps*, UIST '01: PROCEEDINGS OF THE 14TH ANNUAL ACM SYMPOSIUM ON USER INTERFACE SOFTWARE AND TECHNOLOGY, Nov. 2001, at 71–80, <https://doi.org/10.1145/502348.502359>.

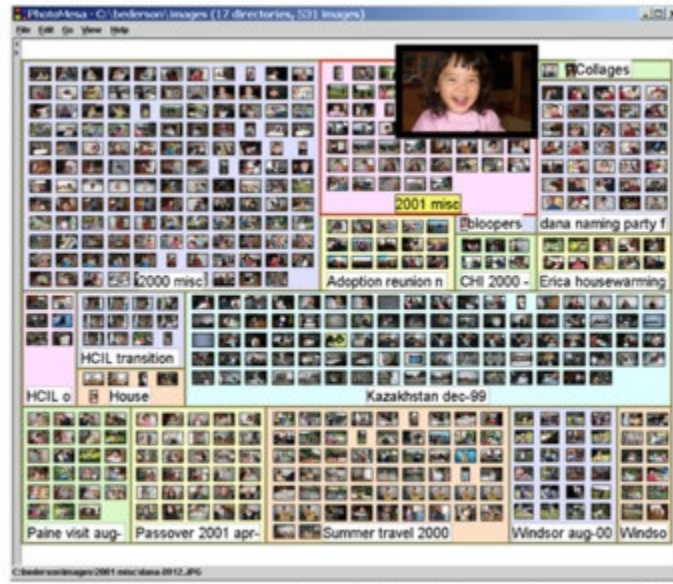


Figure 1: Screen snapshot of PhotoMesa with over 500 images in 17 groups.

180. One of the reasons for designing the interface with zooming was to support devices with different screen sizes. A 2004 paper⁸ describes a later version of PhotoMesa that I co-created called “Pocket PhotoMesa” that adapted the image browser to a Personal Digital Assistant (PDA) display:

⁸Amir Khella and Benjamin B. Bederson, *Pocket PhotoMesa: a Zoomable Image Browser for PDAs*, MUM '04: PROCEEDINGS OF THE 3RD INTERNATIONAL CONFERENCE ON MOBILE AND UBIQUITOUS MULTIMEDIA, Oct. 2004, at 19–24, <https://doi.org/10.1145/1052380.1052384>.



Figure 3: Three views of Pocket PhotoMesa at different zoom levels (www.photomesa.com).

181. LaunchTile led to my creation of Zumobi in 2006, where I was responsible for investigating new software platforms and developing new user interface designs that provide efficient and engaging interfaces to permit end users to access a wide range of content on mobile platforms (including the iPhone and Android-based devices). For example, I designed and implemented software called “Ziibii,” a “river” of news for iPhone, software called “ZoomCanvas,” a zoomable user interface for several iPhone apps, and iPhone apps including “Inside Xbox” for Microsoft and Snow Report for REI. At the International Children’s Digital Library (ICDL), I have since 2002 been the technical director responsible for the design and implementation of the web site, www.childrenslibrary.org (originally at www.icdlbooks.org). In particular, I have been closely involved in designing the user interface as well as the software architecture for the web site since its inception in 2002.

182. The ICDL is a collection of about 4,500 freely available children's books from all over the world in over 75 languages. I was closely involved in the design and development of the underlying database and search engine for the ICDL. This included everything from the software stack, the underlying database schema, the supported search queries as well as the implementation of the Graphical User Interface.

183. The ICDL GUI was designed for children to be able to construct searches consisting of combining multiple query terms including pre-defined metadata as well as open keywords. The GUI, as illustrated in the screenshot below included buttons around the periphery. When multiple buttons were pressed (or a keyword was entered), those terms would be ANDed together, sent to the server to perform the query, and the results were displayed in the center where the covers of the books were displayed, eight at a time. Pressing the right arrow used AJAX web technology to dynamically retrieve and display the next 8 book covers from the server without having to reload the entire page. We switched from a similar-looking version of this interface to this AJAX-based approach in about 2006.



Screenshot taken February 19, 2017 by Benjamin B. Bederson from <http://www.childrenslibrary.org/icdl/SimpleSearchCategoryids=25&langid=&pnum=1&cnum=1&text=&lang=English&ilang=English>

184. Beginning in the mid-1990s, I have been responsible for the design and implementation of numerous other web sites in addition to the ICDL. For example, I designed and built my own professional web site when I was an Assistant Professor of Computer Science at the University of New Mexico in 1995 and have continued to design, write the code for, and update both that site (which I moved to the University of Maryland in 1998, currently at

<http://www.cs.umd.edu/~bederson/>) as well as numerous project web sites, such as Pad++, <http://www.cs.umd.edu/hcil/pad++/>. I received the Janet Fabri Memorial Award for Outstanding Doctoral Dissertation for my Ph.D. work in robotics and computer vision. I have combined my hardware and software skills throughout my career in Human-Computer Interaction research, building various interactive electrical and mechanical systems that couple with software to provide an innovative user experience.

185. During a 5-year period starting in about 2002, I collaborated on a European Union funded project to understand family and social needs around communications and developed technologies to support them. In that work, we developed the concept of “Technology Probes” to help develop this understanding – and that work resulted in a 2003 paper⁹ that, according to Google Scholar¹⁰, was cited over 1,750 times. This early exploratory work resulted in our development of technology to support shared family calendars described in a 2006 paper¹¹ as

⁹ Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B. Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, Nicolas Roussel, and Björn Eiderbäck. 2003. Technology probes: inspiring design for and with families. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03). Association for Computing Machinery, New York, NY, USA, 17–24. <https://doi.org/10.1145/642611.642616>

¹⁰ <https://scholar.google.com/citations?user=IcW1vJEAAAAAJ&hl=en&oi=ao>

¹¹ Catherine Plaisant, Aaron Clamage, Hilary Browne Hutchinson, Benjamin B. Bederson, and Allison Druin. 2006. Shared family calendars: Promoting symmetry

illustrated in Figure 1 below. That work focused on enabling intergenerational and distributed families to have access to information regarding activities of each other. It allowed not only younger family members to support their parents and grandparents, but also supported older adults to gain an understanding on what was going on in the lives of younger family members.

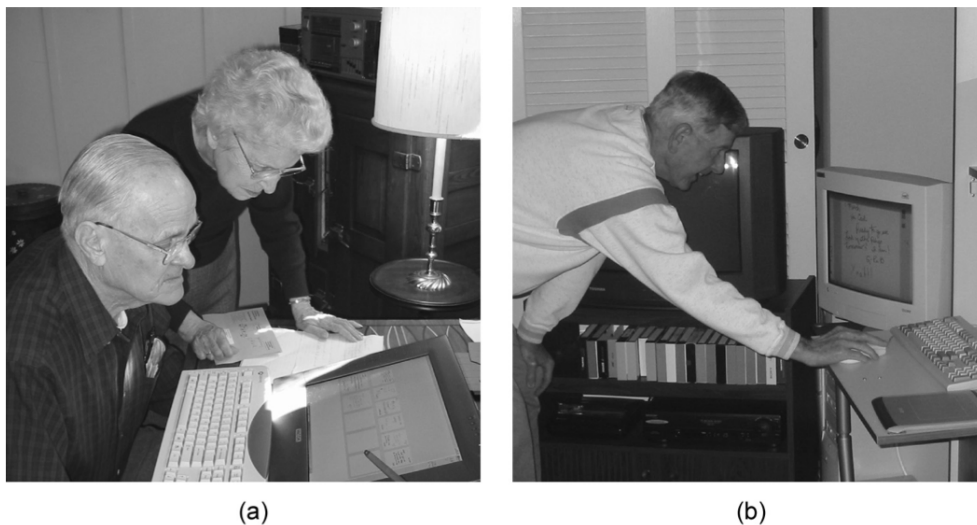


Fig. 1. Using the MessageProbe in the homes of the U.S. family. The MessageProbe was used in the early stages of the project to inspire our design partners to think of new technologies for families.

186. Starting in about 2010, I worked on a National Science Foundation funded project on “human computation” that augmented traditional machine learning by combining human effort with automated systems to translate text. Thus resulted in higher quality text than machine translation systems could do by themselves while working with monolingual participants (which are more available

and accessibility. ACM Trans. Comput.-Hum. Interact. 13, 3 (September 2006), 313–346. <https://doi.org/10.1145/1183456.1183458>

and less expensive than bilingual people). This work, described in a 2012 paper¹² and illustrated in Figure 12 below, used computer networks with crowds of people working on client computers connected to servers to distribute and collect the work of those people to produce the translations.

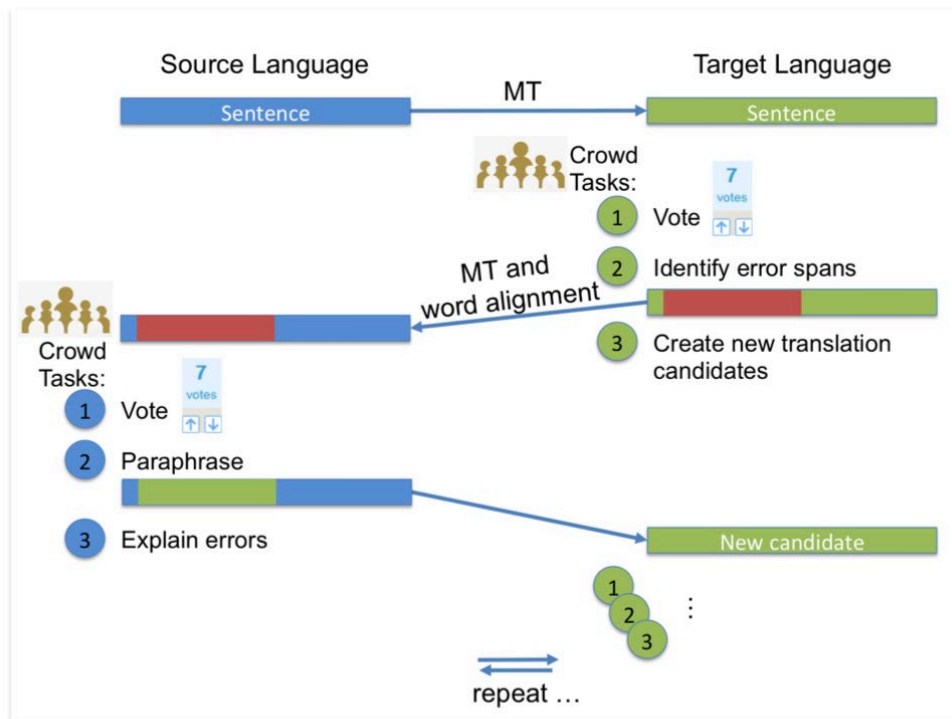


Fig. 12. Collaborative protocol for monolingual translation.

187. My work has been published extensively in more than 160 technical publications, and I have given about 100 invited talks, including 9 keynote lectures. I have won a number of awards including the Brian Shackel Award for

¹² Philip Resnik, Olivia Buzek, Yakov Kronrod, Chang Hu, Alexander J. Quinn, and Benjamin B. Bederson. 2013. Using targeted paraphrasing and monolingual crowdsourcing to improve translation. *ACM Trans. Intell. Syst. Technol.* 4, 3, Article 38 (June 2013), 21 pages. <https://doi.org/10.1145/2483669.2483671>

“outstanding contribution with international impact in the field of HCI” in 2007, and the Social Impact Award in 2010 from Association for Computing Machinery’s (“ACM”) Special Interest Group on Computer Human Interaction (“SIGCHI”). ACM is the primary international professional community of computer scientists, and SIGCHI is the primary international professional HCI community. I have been honored by both professional organizations. I am an “ACM Distinguished Scientist,” which “recognizes those ACM members with at least 15 years of professional experience and 5 years of continuous Professional Membership who have achieved significant accomplishments or have made a significant impact on the computing field.” I am a member of the “CHI Academy,” which is described as follows: “The CHI Academy is an honorary group of individuals who have made substantial contributions to the field of human-computer interaction. These are the principal leaders of the field, whose efforts have shaped the disciplines and/or industry, and led the research and/or innovation in human-computer interaction.” The criteria for election to the CHI Academy are: (1) cumulative contributions to the field; (2) impact on the field through development of new research directions and/or innovations; and (3) influence on the work of others. I have received two “Test of Time” awards from IEEE InfoVis

for my 2001 and 2002 work on visualizing hierarchies, including those containing digital photos.¹³

188. I have appeared on radio shows numerous times to discuss issues relating to user interface design and people's use and frustration with common technologies, web sites, and mobile devices. My work has been discussed and I have been quoted by mainstream media around the world over 120 times, including by the New York Times, the Wall Street Journal, the Washington Post, Newsweek, the Seattle Post-Intelligencer, the Independent, Le Monde, NPR's *All Things Considered*, New Scientist Magazine, and MIT's Technology Review.

189. I have designed, programmed, and publicly deployed dozens of user-facing software products that have cumulatively had millions of users. I am the co-inventor of 14 U.S. patents and 20 U.S. patent applications.

A. Compensation

190. For my efforts in connection with the preparation of this declaration I have been compensated at my standard rate for this type of consulting activity. My compensation is in no way contingent on the results of these or any other proceedings relating to the above-captioned patent.

¹³ <https://ieevis.org/year/2021/info/awards/test-of-time-awards>;
<https://ieevis.org/year/2022/info/awards/test-of-time-awards>.

B. Materials and Other Information Considered

191. I have considered information from various sources in forming my opinions. I have reviewed and considered each of the exhibits listed in the attached Appendix 2 (Materials Considered in the Preparation of This Declaration) in forming my opinions.

VI. UNDERSTANDING OF THE LAW

192. I am not an attorney. In forming my opinions in this Declaration, I applied the relevant legal principles provided to me by counsel, which are summarized in Appendix 4.

VII. RESERVATION OF RIGHTS

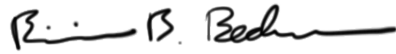
193. My opinions are based upon the information that I have considered to date. I am unaware of any evidence of secondary considerations with respect to the '741 Patent that would render any of the challenged claims non-obvious. I reserve the right, however, to supplement my opinions in the future to respond to any arguments raised by the owner of the '741 Patent and to take into account new information that becomes available to me.

194. I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements

and the like so made are punishable by fine or imprisonment, or both, under
Section 1001 of Title 18 of the United States Code.

Executed on June 25, 2024.

By:

A handwritten signature in black ink, appearing to read "B. B. Bederson", written over a horizontal line.

Benjamin B. Bederson, Ph.D.

**APPENDIX 1: CURRICULUM VITAE OF BENJAMIN B. BEDERSON,
PH.D.**

Dr. Benjamin B. Bederson

University of Maryland, College Park
Professor Emeritus, Computer Science

bederson@umd.edu
www.cs.umd.edu/~bederson

I have read the following and certify that it is a current and accurate statement of my professional record as of May 15, 2024.



1. PERSONAL INFORMATION

EDUCATION

9/89 – 6/92	Ph.D., New York University, Computer Science Dept.	New York, NY
9/87 – 8/89	M.S., New York University, Computer Science Dept.	New York, NY
9/82 – 6/86	B.S., Rensselaer Polytechnic Institute, Computer Science Dept. Minor, Electrical Engineering	Troy, NY

EMPLOYMENT

3/22 – Present	J.S. Bryant School, Inc. Therapeutic high school for LGBTQ+ students Co-founder, Chief Operating Officer, Board member, Treasurer and Clerk	Cummington, MA
1/19 – Present	Bederson Expert Services, Inc. Technical consulting	Windsor, MA
1/98 – Present	University of Maryland, Computer Science Dept./UMIACS 7/19 – Present: Professor Emeritus, Computer Science Department. 4/14 – 6/18: Associate Provost of Learning Initiatives, and Executive Director of Teaching and Learning Transformation Center 4/13 – 4/14: Special Advisor to the Provost on Technology and Educational Transformation 8/11 – 6/19: Professor 9/04 – 7/11: Associate Professor 6/00 – 9/06: Director of the Human-Computer Interaction Lab (HCIL) 1/98 – 8/04: Assistant Professor	College Park, MD
1/06 – Present	International Children's Digital Library Foundation Co-founder and Technical Director for this library of exemplary free online children's books from around the world. 7/17 – present: President, secretary and treasurer	Windsor, MA
10/14 – 4/23	Hazel Analytics, Inc. Co-founder & Board member: regulatory data analytics 10/14 – 12/19: Chief Technology Officer	Seattle, WA
9/06 – 10/14	Zumobi, Inc. Co-founder of VC-funded startup for mobile apps and advertising. 9/06–1/08: VP Client Technologies 1/08–10/14: Chief Scientist	Seattle, WA
6/03 – 11/08	Windsor Interfaces, Inc. Founder: Small company to commercialize HCIL products.	Chevy Chase, MD
1/95 – 8/97	University of New Mexico, Computer Science Department Assistant Professor in the Computer Science Department.	Albuquerque, NM
9/92 – 8/94	Bell Communications Research (Bellcore) Research Scientist in Computer Graphics and Interactive Media group.	Morristown, NJ

6/93 – 7/94	New York University, Media Research Laboratory Visiting Research Scientist.	New York, NY
9/90 – 8/92	Vision Applications, Inc. Research Scientist.	New York, NY
9/88 – 6/90	New York University, Computer Science Department Teaching Assistant.	New York, NY
1/87 – 5/87	North Slope Borough/North Slope Higher Education Center Instructor. Taught Computers & Society to rural Eskimo population.	Barrow, AK

2. RESEARCH, SCHOLARLY, AND CREATIVE ACTIVITIES

Note about publications:

- In all references, my name is in bold.
- Authors are displayed as published. Authors are usually listed in order of decreasing contribution – except that sometimes the last author is the lead advising faculty member. I put an asterisk next to my name when I served that role.
- Students that I supervised are underlined.

A. BOOKS

i. Authored Books

1. Herrnson, P.S., Niemi, R.G., Hanmer, M.J., **Bederson, B. B.**, Conrad, F.G., Traugott, M., (2008) “Voting Technology and the Not-So-Simple Act of Casting a Ballot,” *Brookings Institute Press*.

ii. Edited Collections

1. Teevan, J., Jones, W., **Bederson, B. B.** (Eds.) (2006) *Communications of the ACM (CACM)*, Editors of special issue on “Personal Information Management”, 49 (1), ACM Press.
2. **Bederson, B. B.**, Shneiderman, B. (Eds.) (2003). *The Craft of Information Visualization: Readings and Reflections*. San Francisco: Morgan Kaufmann.

iii. Book Chapters

1. Fails, J., Druin, A., **Bederson, B. B.**, Weeks, A., & Rose, A. (2009) A Child’s Mobile Digital Library: Collaboration, Community and Change. Druin, A. (Ed.) *Mobile Technology for Children: Designing for Interaction and Learning*, Morgan Kaufmann Press.
2. Herrnson, P.S., Niemi, R.G., Hanmer, M.J., **Bederson, B. B.**, Conrad, F.G., Traugott, M.W. (2008) The Current State of Electronic Voting in the United States. Chen, H., Brandt, L., Dawes, S., Gregg, V., Hovy, E., Macintosh, A., Traunmüller, R., Larson, C. (Eds.) *Digital Government: Advanced Research and Case Studies*, Springer, 157-180.
3. Karlson, A., **Bederson, B. B.**, & Contreras-Vidal, Jose L. (2007) Understanding One Handed Use of Mobile Devices. Lumsden, Jo (Ed.), *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, Idea Group Reference, 86-101.
4. Hutchinson, H., Druin, A., **Bederson, B. B.** (2007) Designing Searching and Browsing Software for Elementary-Age Children. Lazar, J. (Ed.), *Universal Usability*. John Wiley. 1618-1630.
5. Druin, A., **Bederson, B. B.**, Boltman, A., Muira, A., Knotts-Callahan, D., & Platt, M. (1999). Children As Our Technology Design Partners. A. Druin (Ed.), *The Design of Children's Technology* (51-72). San Francisco: Morgan Kaufmann.
6. Hollan, J. D., **Bederson, B. B.**, & Helfman, J. (1998). Information Visualization. Helander, M., Landauer, T. K., & Prabhu, P. V. (eds.), *The Handbook of Human Computer Interaction* (Chap. 2, 33-48). Amsterdam: Elsevier Press.

7. **Bederson, B. B.**, Hollan, J. D., Stewart, J., Rogers, D., Druin, A., Vick, D., Ring, L., Grose, E., & Forsythe, C. (1997). A Zooming Web Browser. C. Forsythe, J. Ratner, & E. Grose (eds.), *Human Factors and Web Development* (Chap. 19, 255-266). New Jersey: Lawrence Erlbaum.
8. **Bederson, B. B.**, & Druin, A. (1995). Computer Augmented Environments: Physical Spaces to Enrich Our Lives. Jakob Nielsen (eds.), *Advances in Human-Computer Interaction* (Vol. 5, Chap. 2, 37-66). New Jersey: Ablex Press.
9. **Bederson, B. B.**, Wallace, R. S., & Schwartz, E. L. (1993). A Miniaturized Space-Variant Active Vision System: Cortex-I. Richard Mammone (eds.), *Artificial Neural Networks for Speech and Vision*. Chapman-Hall Publishers.
10. Wallace, R. S., Ong, P.-W., **Bederson, B. B.**, & Schwartz, E. L. (1993). Connective Graphs in Space-Variant Active Vision. George A. Bekey & Ken Goldberg (eds.), *Neural Networks in Robotics*. Kluwer Academic Publishers.

B. REFEREED ARTICLES

i. Journals

1. **Bederson, B. B.**, Jin, G. Z., Leslie, P., Quinn, A. J., Zou, B. (2017) "Incomplete Disclosure: Evidence of Signaling and Countersignaling", in *American Economic Journal: Microeconomics*, (in press).
2. Anderson, L. B., Gardner, E. E., Wolvin, A., Kirby-Straker, R., Yalcin, A., & **Bederson, B. B.*** (2016), "Incorporating Learning Analytics into Basic Course Administration: How to Embrace the Opportunity to Identify Inconsistencies and Inform Responses", in *Journal of the Association for Communication Administration*, 35 (1), Winter-Spring 2016, pp 2-14.
3. Lee, T. Y., **Bederson, B. B.*** (2016) "Give the people what they want: studying end-user needs for enhancing the web", *PeerJ Computer Science*, Nov, 2016, 2:e91 <https://doi.org/10.7717/peerj-cs.91>
4. Yalcin, M. A., Elmqvist, N. & **Bederson, B. B.***. (2015) "AggreSet: Rich and Scalable Set Exploration using Visualizations of Element Aggregations". in *IEEE Transactions on Visualization and Computer Graphics*, 22(1), 688–697, 2015.
5. Hu, C., Resnik, P., & **Bederson, B. B.*** (2014) "Crowdsourced Monolingual Translation", in *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21 (4), Aug 2014, pp. 22:1–22:35.
6. Lee, T. Y., Mauriello, M., Ahn, J. & **Bederson, B. B.*** (2014) "CTArcade: Computational Thinking with Games in School Age Children", in *International Journal of Child-Computer Interaction*, 2 (1), Jan 2014, 26–33.
7. Bonsignore, E., Quinn, A. J., Druin, A. & **Bederson, B. B.*** (2013) "Sharing Stories 'In The Wild': A Mobile Storytelling Case Study using StoryKit", in *Transactions on Computer-Human Interaction (TOCHI)*, ACM, 20 (3), July 2013, Article 18.
8. Resnik, P., Buzek, O., Kronrod, Y., Hu, C., Quinn, A. J. & **Bederson, B. B.** (2013) "Using Targeted Paraphrasing and Monolingual Crowdsourcing to Improve Translation", in *Transactions on Intelligent Systems and Technology*, ACM, 4 (3), July 2013, Article 38, 21 pages.
9. **Bederson, B. B.** (2011) "The Promise of Zoomable User Interfaces", in *Behaviour & Information Technology*, Taylor & Francis, Nov-Dec 2011, 30 (6), 853-866.
10. Druin, A., **Bederson, B. B.**, Rose, A., Weeks, A. (2009) "From New Zealand to Mongolia: Co-Designing and Deploying a Digital Library for the World's Children", in *Children, Youth and Environments*, University of Colorado, 19 (1), 34-57.
11. Conrad, F.G., **Bederson, B. B.**, Lewis, B., Peytcheva, E., Traugott, M.W., Hanmer, M.J., Herrnson, P.S., & Niemi, R.G. (2009) "Electronic voting eliminates hanging chads but introduces new usability challenges", in *International Journal of Human-Computer Studies (IJHCS)*, Elsevier Press, 67 (1), 111-124.
12. Hanmer, M.J., Park, W.-H., Traugott, M.W., Niemi, R.G., Herrnson, P.S., **Bederson, B. B.**, and Conrad, F.C. (2008) "Losing Fewer Votes: The Impact of Changing Voting Systems on Residual Votes", in *Political Research Quarterly*, Sage Publications, September, 2008. doi:10.1177/1065912908324201.

13. **Bederson, B. B.**, Clamage, A., Plaisant, C. (2008) Enhancing In-Car Navigation Systems with Personal Experience, in *Transportation Research Record (TRR)*, The National Academies, 2064 (2008) 33-42.
14. Herrnson, P.S., Niemi, R.G., Hanmer, M.J., Francia, P.L., **Bederson, B. B.**, Conrad, F.G., Traugott, M. (2008) Voter's Evaluations of Electronic Voting Systems: Results from a Usability Field Study, *American Politics Research*, Sage Journals, 36 (4), July 2008, 580-611.
15. Cockburn, A., Karlson, A., & **Bederson, B. B.*** (2008) A Review of Overview+Detail, Zooming, and Focus+Context Interfaces, *ACM Computing Surveys*, ACM Press, December 2008, 41 (1).
16. Kang, H., **Bederson, B. B.**, & Suh, B. (2008) Capture, Annotate, Browse, Find, Share: Novel Interfaces for Personal Photo Management, *International Journal of Human-Computer Interaction (IJHCI)*, 23 (3), 1-23.
17. Parr, C.S., Lee, B., **Bederson, B. B.*** (2007) EcoLens: Integration and Interactive Visualization of Ecological Datasets, *Journal of Ecological Informatics*, Elsevier, 2 (1), 61-69.
18. Suh, B., and **Bederson, B. B.*** (2007) Semi-Automatic Photo Annotation Strategies Using Event Based Clustering and Clothing Based Person Recognition, *Interacting With Computers*, Elsevier, 19 (4), 524-544.
19. Hutchinson, H., **Bederson, B. B.**, Druin, A., (2007) Supporting Elementary-Age Children's Searching and Browsing: Design and Evaluation Using the International Children's Digital Library, *Journal of the American Society for Information Science and Technology*, John Wiley & Sons, 58 (11), 1618-1630.
20. Kang, H., Plaisant, C., Lee, B., **Bederson, B. B.***, (2007) NetLens: Iterative Exploration of Content-Actor Network Data, *Information Visualization*, Palgrave Macmillan, 6 (1), 18-31.
21. Lee, B., Parr, C.S., Plaisant, C., **Bederson, B. B.**, Veksler, V.D., Gray, W.D., Kotfila, C. (2006) TreePlus: Interactive Exploration of Networks with Enhanced Tree Layouts, *IEEE Transactions on Vision and Computer Graphics (TVCG)*, IEEE Press, 12 (6), 1414-1426.
22. Plaisant, C., **Bederson, B. B.**, Clamage, A., Hutchinson, H.B., Druin, A. (2006) Shared Family Calendars: Promoting Symmetry and Accessibility, *Transactions on Computer-Human Interaction*, New York: ACM, 13 (3), 313-346.
23. Shneiderman, B., **Bederson, B. B.**, and Drucker, S., (2006) Find that photo! Interface strategies to annotate, browse, and share, *Communications of the ACM*, 49 (4), 69-71.
24. Hutchinson, H.B., Rose, A., **Bederson, B. B.**, Weeks, A.C., Druin, A. (2005) The International Children's Digital Library: A Case Study in Designing for a Multi-Lingual, Multi-Cultural, Multi-Generational Audience, *Information Technology and Libraries*, 24 (1), 4-12.
25. Herrnson, P.S., **Bederson, B. B.**, Lee, B., Francia, P. L., Sherman, R.M., Conrad, F.G., Traugott, M., Niemi, R.G. (2005) Early Appraisals of Electronic Voting, *Social Science Computing Review*, 23 (3), 274-292.
26. Hourcade, J. P., **Bederson, B. B.**, Druin, A., & Guimbretière, F. (2004) Differences in pointing task performance between preschool children and adults using mice, *Transactions on Computer-Human Interaction*, New York: ACM, 11 (4), 357-386.
27. Parr, C. S., Lee, B., & **Bederson, B. B.*** (2004). Visualizations for Taxonomic and Phylogenetic Trees. *Bioinformatics*, Oxford University Press, 20 (17), 2997-3004.
28. **Bederson, B. B.**, Grosjean, J., Meyer, J. (2004) Toolkit Design for Interactive Structured Graphics, *Transactions on Software Engineering*, New York: IEEE, 30 (8), 535-546.
29. Chipman, L. E., **Bederson, B. B.**, Golbeck, J. A. (2004) SlideBar: Analysis of a Linear Input Device. *Journal of Behaviour and Information Technology*, London, Taylor & Francis, 23 (1), 1-9.
30. **Bederson, B. B.**, Clamage, A., Czerwinski, M. P., & Robertson, G. G. (2004) DateLens: A Fisheye Calendar Interface for PDAs, *Transactions on Computer-Human Interaction*, New York: ACM, 11 (1), 90-119.
31. Hourcade, J. P., **Bederson, B. B.**, & Druin, A. (2004) Building KidPad: An Application for Children's Collaborative Storytelling, *Software: Practice & Experience*, New Jersey: John Wiley & Sons, Ltd, 34 (9), 895-914.

32. Druin, A., **Bederson, B. B.**, Weeks, A., Farber, A., Grosjean, J., Guha, M. L., Hourcade, J. P., Lee, J., Liao, S., Reuter, K., Rose, A., Takayama, Y., & Zhang, L. (2003). The International Children's Digital Library: Description and Analysis of First Use. *First Monday*, 8(5).
33. Druin, A., Reville, G., **Bederson, B. B.**, Hourcade, J. P., Farber, A., Lee, J., & Campbell, D. (2003) A Collaborative Digital Library for Children: A Descriptive Study of Children's Collaborative Behavior and Dialogue. *The Journal of Computer-Assisted Learning*, 19 (2), 239-248.
34. Hourcade, J.P., **Bederson, B. B.**, Druin, A., Rose, A., Farber, A., Takayama, Y. (2003). The International Children's Digital Library: Viewing Digital Books Online. *Interacting with Computers*, Elsevier Press, 15 (3), 151-167.
35. Hornbæk, K., **Bederson, B. B.**, & Plaisant, C. (2002) Navigation Patterns and Usability of Zoomable User Interfaces With and Without an Overview, *ACM Transactions on Computer-Human Interaction*, 9 (4), 362-389.
36. **Bederson, B. B.**, Shneiderman, B., & Wattenberg, M. (2002) Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies, *ACM Transactions on Graphics*, 21 (4), 833-854, ACM Press. – **Won 2021 IEEE Vis Test of Time Award in InfoVis**, <http://ieevis.org/year/2021/info/awards/test-of-time-awards>.
37. Good, L., & **Bederson, B. B.*** (2002) Zoomable User Interfaces as a Medium for Slide Show Presentations, *Information Visualization*, Palgrave Macmillan, 35-49.
38. Reville, G., Druin, A., Platner, M., **Bederson, B. B.**, Hourcade, J.P., & Sherman, L. (2002) A Visual Search Tool for Early Elementary Science Students, *Journal of Science Education and Technology*, 11(1), 49-57.
39. **Bederson, B. B.**, & Meyer, J. (1998). Implementing a Zooming User Interface: Experience Building Pad++. *Software: Practice and Experience*, 28(10), 1101-1135.
40. **Bederson, B. B.**, Hollan, J. D., Perlin, K., Meyer, J., Bacon, D., & Furnas, G. W. (1996). Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics. *Journal of Visual Languages and Computing*, 7, 3-31.
41. **Bederson, B. B.**, Wallace, R. S., & Schwartz, E. L. (1995). A Miniaturized Space-Variant Active Vision System: Cortex-I. *Machine Vision and Applications*, 8 (2), 101-109.
42. **Bederson, B. B.**, Wallace, R. S., & Schwartz, E. L. (1994). A Miniature Pan-Tilt Actuator: the Spherical Pointing Motor. *IEEE Transactions on Robotics and Automation*, 10(3), 298-308.
43. Wallace, R. S., Ong, P.-W., **Bederson, B. B.**, & Schwartz, E. L. (1994). Space-Variant Image Processing. *International Journal of Computer Vision*, 13(1), 71-90.

ii. Peer-Reviewed Published Full-Length Conference Papers

1. Yalçın, M. A., Elmqvist, N., & **Bederson, B. B.*** (2017) Raising the Bars: Evaluating Treemaps vs. Wrapped Bars for Dense Visualization of Sorted Numeric Data, *Proceedings of Graphics Interface (GI 2017)*, 41-49. DOI: <https://doi.org/10.20380/GI2017.06>
2. Quinn, A., & **Bederson, B. B.*** (2014) AskSheet: Efficient Use of Web Workers to Support Decision Making, *Proceedings of ACM CSCW (CSCW 2014)*, ACM Press, pp. 1456-1466.
3. Yeh, T., Chang, T.-H., Xie, B., Walsh, G., Watkins, I., Wongsuphasawat, K., Davis, L.S., & **Bederson, B. B.*** (2011) Creating Contextual Help for GUIs Using Screenshots, *Proceedings of ACM UIST (UIST 2011)*, ACM Press, 145-154.
4. Quinn, A., & **Bederson, B. B.*** (2011) Human Computation: A Survey and Taxonomy of a Growing Field, *Proceedings of ACM CHI (CHI 2011)*, ACM Press, 1403-1412.
5. Resnik, P., Buzek, O., Hu, C., Kronrod, Y., Quinn, A.J., & **Bederson, B. B.*** (2010) Improving Translation via Targeted Paraphrasing, *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP 2010)*, 127-137.
6. Hu, C., **Bederson, B. B.**, & Resnik, P. (2010) Translation by Interactive Collaboration between Monolingual Users, *Proceedings of Graphics Interface (GI 2010)*, 39-46.

7. Dearman, D., Karlson, A., Meyers, B., & **Bederson, B. B.** (2010) Multi-Modal Text Entry and Selection on Mobile Devices, *Proceedings of Graphics Interface (GI 2010)*, 19-26.
8. Quinn, A., Hu, C., Arisaka, T., & **Bederson, B. B.*** (2008) Readability of Scanned Books in Digital Libraries, *Proceedings of ACM CHI (CHI 2008)*, ACM Press, 705-714. [22% acceptance rate]
9. Karlson, A., & **Bederson, B. B.*** (2008) One-Handed Touchscreen Input for Legacy Applications, *Proceedings of ACM CHI (CHI 2008)*, ACM Press 1399-1408. [22% acceptance rate]
10. **Bederson, B. B.**, Clamage, A., Plaisant, C., (2008) Enhancing In-Car Navigation Systems with Personal Experience, in *Proc. of the Transportation Research Board 87th annual meeting*, The National Academies, Washington, DC (2008) 1-11.
11. Karlson, A., & **Bederson, B. B.*** (2007) ThumbSpace: Generalized One-Handed Input for Touchscreen-Based Mobile Devices, *Proceedings of INTERACT 2007*, 324-338 – **Winner of Brian Shackel Award for making an “outstanding contribution with international impact in the field of HCI”.**
12. Druin, A., Weeks, A., Massey, S., & **Bederson, B. B.** (2007) Children’s Interests and Concerns When Using the International Children’s Digital Library: A four country case study, *Proceedings of the Joint Conference on Digital Libraries (JCDL 2007)*, 167-176. – **Nominated for Best Paper Award, came in #2.** [36% acceptance rate].
13. Parhi, P., Karlson, A., & **Bederson, B. B.*** (2006). “Target Size Study for One-Handed Thumb Use on Small Touchscreen Devices”, *Proceedings of MobileHCI 2006*, ACM Press, 203-210. – **Winner of Best Historical Paper Award, September 2016.**
14. Hutchinson, H. B., **Bederson, B. B.**, & Druin, A. (2006). “The Evolution of the International Children’s Digital Library Searching and Browsing Interface”, *Proceedings of 5th International Conference for Interaction Design and Children (IDC 2006)*, ACM Press, 105-112. [38% acceptance rate]
15. Hutchinson, H., Druin, A., **Bederson, B. B.**, Reuter, K., Rose, A., Weeks, A. (2005) How do I Find Blue Books About Dogs? The Errors and Frustrations of Young Digital Library Users, *Proceedings of HCI 2005*, Las Vegas, NV (CD-ROM).
16. Shneiderman, B. & **Bederson, B. B.** (2005). “Maintaining Concentration to Achieve Task Completion”, *Proceedings of Designing User Experience (DUX 2005)*.
17. Karlson, A., **Bederson, B. B.**, & SanGiovanni, J. (2005). AppLens and LaunchTile: Two Designs for One-Handed Thumb Use on Small Devices, *CHI Letters*, 7(1), 201-210. [25% acceptance rate] – **Nominated for Best Paper Award.**
18. Khella, A., **Bederson, B. B.*** (2004). Pocket PhotoMesa: A Zoomable Image Browser for PDAs. In *Proceedings of Mobile and Ubiquitous Multimedia (MUM 2004)* ACM Press, 19-24.
19. Lee, B., Parr, C. S., Campbell, D., & **Bederson, B. B.*** (2004). How Users Interact With Biodiversity Information Using TaxonTree. In *Proceedings of Advanced Visual Interfaces (AVI 2004)* ACM Press, 320-327. [25% acceptance rate]
20. Suh, B., Ling, H., **Bederson, B. B.**, & Jacobs, J. W. (2003). Automatic Thumbnail Cropping and Its Effectiveness. *UIST 2003, ACM Symposium on User Interface Software and Technology*, *CHI Letters*, 5(2), 95-104. [22% acceptance rate] – **Winner of Best Student Paper Award.**
21. **Bederson, B. B.**, Lee, B., Sherman, R., Herrnson, P. S., Niemi, R. G. (2003). Electronic Voting System Usability Issues. *CHI 2003, ACM Conference on Human Factors in Computing Systems*, *CHI Letters*, 5(1), 145-152. [16% acceptance rate]
22. Hutchinson, H., Mackay, W., Westerlund, B., **Bederson, B. B.**, Druin, A., Plaisant, C., Evans, H., Hansen, H., Conversy, S., Beaudouin-Lafon, M., Roussel, N., Lacomme, L., Eiderbäck, B., Lindquist, S., Sundblad, Y. (2003). Technology Probes: Inspiring Design for and with Families. *CHI 2003, ACM Conference on Human Factors in Computing Systems*, *CHI Letters*, 5(1), 17-24. [16% acceptance rate]
23. Knudtson, K., Druin, A., Kaplan, N., Summers, K., Chisik, Y., Kulkarni, R., Moulthrop, S., Weeks, H., & **Bederson, B.** Starting an Intergenerational Technology Design Team: A Case Study. In *Proceedings of Interaction Design and Children (IDC 2003)* ACM Press, 51-58.

24. Baudisch, P., Cutrell, P., Robbins, D., Czerwinski, M., Tandler, P., **Bederson, B. B.**, & Zierlinger, A. (2003) Drag-and-Pop and Drag-and-Pick: Techniques for Accessing Remote Screen Content on Touch- and Pen-Operated Systems. *Proceedings of Interact 2003*, Austria: International Federation for Information Processing, 57-64. [33% acceptance rate]
25. Plaisant, C., Grosjean, J., & **Bederson, B. B.*** (2002) SpaceTree: Design Evolution of a Node Link Tree Browser. *Proceedings of Information Visualization Symposium (InfoVis 2002)* New York: IEEE, 57-64. [27% acceptance rate] – **Won 2022 IEEE Vis Test of Time Award in InfoVis**, <https://ieevis.org/year/2022/info/awards/test-of-time-awards>.
26. Suh, B., & **Bederson, B. B.*** (2002) OZONE: A Zoomable Interface for Navigating Ontology Information. *Proceedings of International Conference on Advanced Visual Interfaces (AVI 2002)*, ACM Press, Trento, Italy, 139-143. [30% acceptance rate]
27. Boltman, A., Druin, A., **Bederson, B.**, Hourcade, J.P., Stanton, D., O'Malley, C., Cobb, S., Benford, S., Fast, C., Kjellin, M., & Sundblad, Y. (2002). The nature of children's storytelling with and without technology. *Proceedings of American Educational Research Association (AERA) Conference*, New Orleans.
28. **Bederson, B. B.** (2001). PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps. *UIST 2001, ACM Symposium on User Interface Software and Technology*, CHI Letters, 3(2), 71-80. [19% acceptance rate]
29. Druin, A., **Bederson, B. B.**, Hourcade, J.P., Sherman, L., Reville, G., Platner, M., Weng, S. (2001) Designing a Digital Library for Young Children. In *Proceedings of Joint Conference on Digital Libraries (JCDL 2001)* ACM Press, 398-405. [30% acceptance rate]
30. Benford, S., **Bederson, B. B.**, Åkesson, K., Bayon, V., Druin, A., Hansson, P., Hourcade, J. P., Ingram, R., Neale, H., O'Malley, C., Simsarian, K., Stanton, D., Sundblad, Y., & Taxén, G. (2000). Designing Storytelling Technologies to Encourage Collaboration Between Young Children. *CHI 2000, ACM Conference on Human Factors in Computing Systems*, CHI Letters, 2(1), 556-563. [19% acceptance rate]
31. **Bederson, B. B.**, Meyer, J., & Good, L. (2000). Jazz: An Extensible Zoomable User Interface Graphics Toolkit in Java. *UIST 2000, ACM Symposium on User Interface Software and Technology*, CHI Letters, 2(2), 171-180. [26% acceptance rate]
32. **Bederson, B. B.** (2000). Fisheye Menus. *UIST 2000, ACM Symposium on User Interface Software and Technology*, CHI Letters, 2(2), 217-225. [26% acceptance rate]
33. Gandhi, R., Kumar, G., **Bederson, B. B.**, & Shneiderman, B. (2000). Domain Name Based Visualization of Web Histories in a Zoomable User Interface. In *Proceedings of Eleventh International Workshop on Database and Expert Systems Applications – Second International Workshop on Web-Based Information Visualization (WebVis 2000)*, IEEE Computer Society, Los Alamitos, CA (2000), 591-598.
34. **Bederson, B. B.**, & Boltman, A. (1999). Does Animation Help Users Build Mental Maps of Spatial Information? In *Proceedings of Information Visualization Symposium (InfoVis 99)* New York: IEEE, 28-35. [23% acceptance rate]
35. Combs, T. T. A., & **Bederson, B. B.*** (1999). Does Zooming Improve Image Browsing? In *Proceedings of Digital Library (DL 99)* New York: ACM, 130-137. [21% acceptance rate]
36. Stewart, J., **Bederson, B. B.**, & Druin, A. (1999). Single Display Groupware: A Model for Co-Present Collaboration. In *Proceedings of Human Factors in Computing Systems (CHI 99)* ACM Press, 286-293. [25% acceptance rate]
37. Hightower, R. R., Ring, L., Helfman, J., **Bederson, B. B.**, & Hollan, J. D. (1998). Graphical Multiscale Web Histories: A Study of PadPrints. In *Proceedings of ACM Conference on Hypertext (Hypertext 98)* ACM Press, 58-65.
38. Druin, A., Stewart, J., Proft, D., **Bederson, B. B.**, & Hollan, J. D. (1997). KidPad: A Design Collaboration Between Children, Technologists, and Educators. In *Proceedings of Human Factors in Computing Systems (CHI 97)* ACM Press, 463-470. [22% acceptance rate]

-
39. **Bederson, B. B.**, Hollan, J. D., Stewart, J., Rogers, D., Druin, A., & Vick, D. (1996). A Zooming Web Browser. In *Proceedings of SPIE Multimedia Computing and Networking* New York: IEEE, 260-271. <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/2667/0000/Zooming-Web-browser/10.1117/12.235898.short>
 40. **Bederson, B. B.**, & Hollan, J. D. (1995). Advances in the Pad++ Zoomable Graphics Widget. In *Proceedings of USENIX Tcl/Tk '95 Workshop (Tcl/Tk 95)* New York: USENIX, 3.
 41. Furnas, G. W., & **Bederson, B. B.** (1995). Space-Scale Diagrams: Understanding Multiscale Interfaces. In *Proc. of Human Factors in Computing Systems (CHI 95)* ACM Press, 234-241. [29% acceptance rate]
 42. **Bederson, B. B.**, & Hollan, J. D. (1994). Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. In *Proc. of User Interface and Software Technology (UIST 94)* ACM Press, 17-26.
 43. **Bederson, B. B.**, Wallace, R. S., & Schwartz, E. L. (1993). Control & Design of the Spherical Pointing Motor. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA 93)* New York: IEEE, 2, 630-636.
 44. **Bederson, B. B.**, Wallace, R. S., & Schwartz, E. L. (1992). A Miniaturized Active Vision System. In *Proceedings of Eleventh International Conference on Pattern Recognition*, IEEE Press, 58-61.
 45. **Bederson, B. B.**, Wallace, R. S., & Schwartz, E. L. (1992). Two Miniature Pan-Tilt Devices. In *Proceedings of IEEE International Conference on Robotics and Automation*, New York: IEEE Press, (1), 658-663.
 46. Wallace, R. S., **Bederson, B. B.**, & Schwartz, E. L. (1992). Voice-Bandwidth Visual Communication Through Logmaps: the Telecortex. In *Proceedings of IEEE Workshop on Applications of Computer Vision* New York: IEEE Press, 4-10.

iii. Other Publications

1. Yalcin, M. A., Elmqvist, N., & **Bederson, B. B.*** (2017). Piled Bars: Dense Visualization of Numeric Data. Poster presented at Graphics Interfaces, 2017.
2. **Bederson, B. B.**, Barker, K., Owens, L., Kim, Y. J. & Palmer, K. (2017) Designing Organizations to Support Tech on Campus. Panel at SXSW Edu, Austin, TX, March 6, 2017.
3. Yalcin, M. A., Elmqvist, N., & **Bederson, B. B.*** (2016). Keshif: Out-of-the-Box Visual and Interactive Data Exploration Environment. In *Proceedings of IEEE VIS 2016 Workshop on Visualization in Practice: Open Source Visualization and Visual Analytics Software*.
4. Yalcin, M. A., Elmqvist, N., & **Bederson, B. B.*** (2016). Cognitive Stages in Visual Data Exploration. In *Proceedings of Beyond Time and Errors: Novel Evaluation Methods for Visualization (BELIV 2016)* at IEEE VIS 2016.
5. Roberts, S., Swan, L., & **Bederson, B. B.** (2016). "Fishing for Faculty: The Bait and Hooks", Workshop in 2016 Professional and Organizational Development (POD) Conference, Louisville, KY, November, 2016.
6. Anderson, L. B., Gardner, E. E., Wolvin, A., Kirby-Straker, R., Yalcin, A., & **Bederson, B. B.*** (2015), "Incorporating Learning Analytics into Basic Course Administration: How to Embrace the Opportunity to Identify Inconsistencies and Inform Responses". In *NCA 2015 - 101th Annual meeting of the National Communication Association*. **Top Paper in Basic Course Division**.
7. Anderson, L. B., Gardner, E. E., Wolvin, A., Kirby-Straker, R., Yalcin, A., & **Bederson, B. B.*** (2015), "Utilizing learning analytics to improve the basic course: A qualitative analysis of the instructor feedback captured in Revealer". *Paper presented at the annual meeting of the Eastern Communication Association, Philadelphia, PA, June 2015*.
8. Hara, K., Hajiaghayi, M., & **Bederson, B. B.** (2015). FluTCHA: Using Fluency to Distinguish Humans from Computers, short paper and poster, WWW2015, Florence, Italy, May 18-22, 2015.
9. Teasley, S., & **Bederson, B. B.** (2015). Learning Analytics and iSchools: What, Why & Worries, at *iConference*, Session for Interaction and Engagement, March 24-27, 2015, Newport Beach, CA.

10. Xu, Z., Goldwasser, D., **Bederson, B. B.**, Lin, J. (2014) Visual Analytics of MOOCs at Maryland, *Poster at Learning @ Scale Conference (L@S '14)*.
11. **Bederson, B. B.**, Jin, G., Leslie, P., Quinn, A.J., & Zhou, B. (alphabetical order), (2014) "Incomplete Disclosure: Insights from Maricopa", International Industrial Organization Conference, April 11-13, 2014, Chicago.
12. Lee, T.Y., Mauriello, M.L., Ingraham, J., Sopan, A., Ahn, J., & **Bederson, B. B.*** (2012) "CTArcade: Learning Computational Thinking while Training Virtual Characters Through Game Play", WiP at CHI 2012, ACM Press, 2309-2314.
13. Cranor, L., **Bederson, B. B.**, & Quesenberry, W. (2012) "Occupy CHI! Engaging U.S. Policymakers", Panel in CHI 2012, ACM Press, 1139-1142.
14. **Bederson, B. B.**, Quinn, A. J., & Rose, A. (2012) SearchParty: Learning to Search in a Web-based Classroom, Workshop paper at CHI 2012.
15. Hu, C., Resnik, P., Kronrod, Y., & **Bederson, B. B.*** (2012) Deploying MonoTrans Widgets in the Wild. Note at CHI 2012, ACM Press, 2935-2938.
16. Hu, C., Resnik, P., Kronrod, Y., Eidelman, V., Buzek, O., & **Bederson, B. B.*** (2011) The Value of Monolingual Crowdsourcing in a Real-World Translation Scenario: Simulation using Haitian Creole Emergency SMS Messages. In *Proceedings of EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, 399-404.
17. Kronrad, Y., Hu, C., Buzek, O., & Quinn, A.J. (2011) Using Monolingual Crowd to Improve Translation. Student poster at AAAS 2011. **Winner of poster in Math, Technology and Engineering category.**
18. **Bederson, B. B.**, & Quinn, A.J. (2011) Web Workers Unite! Addressing Challenges of Online Laborers. Alt.CHI paper at CHI 2011, ACM Press, 97-106.
19. Hu, C., **Bederson, B. B.**, & Resnik, P. (2011) MonoTrans2: A New Human Computation System to Support Monolingual Translation. Short paper at CHI 2011, ACM Press, 1133-1136.
20. Kronrod, Y., Buzek, O., Hu, C., Quinn, A., Resnik, P., & **Bederson, B. B.*** (2010) Using Crowdsourcing to Translate Without Bilinguals. Poster at CrowdConf 2010.
21. Hu, C., **Bederson, B. B.**, and Resnik, P. (2010) Translation by Iterative Collaboration Between Monolingual Users. Abstract and demo at the Human Computation Workshop (HCOMP 2010), 54-55.
22. Buzek, O., Resnik, P., & **Bederson, B. B.*** (2010) Error Driven Paraphrase Annotation using Mechanical Turk, In *Creating Speech and Language Data With Amazon's Mechanical Turk*, NAACL 2010 Workshop, 2010.
23. Chen, R., Rose, A., & **Bederson, B. B.*** (2009) How People Read Books Online: Mining and Visualizing Web Logs for Use Information. In *Proceedings of European Conference on Digital Libraries (ECDL 2009)*, Short Paper, 364-369.
24. **Bederson, B. B.**, Rutledge, P., Quinn, A. (2009) Generalizing the International Children's Digital Library. In *Proceedings of the Digital Humanities Conference (DH 09)*, Poster.
25. **Bederson, B. B.**, Quinn, A., Druin, A. (2009) Designing the Reading Experience for Scanned Multi-lingual Picture Books on Mobile Phones. In *Proceedings of the Joint Conference on Digital Libraries (JCDL 2009)*, Short Paper, ACM Press, New York, NY, 305-308.
26. Hu, C., Rose, A., **Bederson, B. B.*** (2009) Locating Text in Scanned Books. In *Proceedings of the Joint Conference on Digital Libraries (JCDL 2009)*, Poster.
27. Druin, A., Cavallo, D., Fabian, C., **Bederson, B. B.**, Revell, G., Rogers, Y. (2009) Mobile Technologies for the World's Children. In *Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2009)*, ACM Press, Panel.
28. **Bederson, B. B.** (2008) "Experiencing the International Children's Digital Library", *Interactions Magazine*, ACM Press, 15 (6), 50-54.
29. Lazar, J., Hochheiser, H., Johnson, J., Karat, C-M., **Bederson, B. B.** (2008) "CHI Policy Issues Around the World", Panel in CHI 2008, ACM Press, 2277-2280.

30. Teevan, J., Jones, W., **Bederson, B. B.**, (2006) "Introduction: Personal Information Management," Communications of the ACM, 49 (1), 40-43. Ranked #3 "Year Rank" in annual downloads among ACM Magazines and Computing Surveys articles (CACM, 50 (12), 92-93).
31. Lee, B., Parr, C.S., Plaisant, C., **Bederson, B. B.**, (2006) "TreePlus: Visualizing Graphs as Trees," Video in InfoVis 2006.
32. Herrnson, P.S., Niemi, R.G., Hanmer, M.J., **Bederson, B. B.**, Conrad, F.G., Traugott, M., (2006) "The Importance of Usability Testing of Voting Systems," in Usenix/Accurate Electronic Voting Workshop.
33. Herrnson, P., Niemi, R., Hanmer, M., **Bederson, B. B.**, Conrad, F., Traugott, M., (2006) "Voter Errors in Electronic voting: Voting Systems, Ballot Type, and Voter Traits," In Midwest Political Science Association annual conference.
34. Herrnson, P., Niemi, R., Hanmer, M., **Bederson, B. B.**, Conrad, F., Traugott, M., (2006) "Voters' Abilities to Cast Write-In Votes Using Electronic Voting Systems," In Midwest Political Science Association annual conference.
35. Conrad, F., Peytcheva, E., Traugott, M., Lewis, B., Herrnson, P., **Bederson, B. B.** (2006) "Usability of Electronic Voting Systems: Field and Laboratory Experiments," In Midwest Political Science Association annual conference.
36. Lee, B., Parr, C. S., Plaisant, C., & **Bederson, B. B.*** (2005) "Visualizing Graphs as Trees: Plant a Seed and Watch it Grow", Proceedings of Graph Drawing 2005, 516-518.
37. Klein, C. & **Bederson, B. B.*** (2005). "Benefits of Animated Scrolling", Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2005) ACM Press, Short Paper, 1965-1968.
38. Lee, B., Czerwinski, M., Robertson, G. G. & **Bederson, B. B.** (2005). "Understanding Research Trends in Conferences using PaperLens", Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2005) ACM Press, Short Paper, 1969-1972.
39. Traugott, M.W., Hanmer, M.J., Park, W.H., Herrnson, P.S., Niemi, R.G., **Bederson, B. B.**, Conrad, F.G.,(2005) "The Impact of Voting Systems on Residual Votes, Incomplete Ballots, and Other Measures of Voting Behavior," In American Political Science Association meeting (APSA 2005).
40. Lee, B., Czerwinski, M., Robertson, G. G. & **Bederson, B. B.** (2004). "Understanding Eight Years of InfoVis Conferences using PaperLens", InfoVis 2004 Symposium Compendium. **Winner of InfoVis Contest.**
41. **Bederson, B. B.** (2004) Interfaces for Staying in the Flow. In Ubiquity, ACM Press, (5) 27.
42. Hourcade, J. P., **Bederson, B. B.**, Druin, A. (2004) Preschool Children's Use of Mouse Buttons. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2004) ACM Press, Short Paper, 1411-1412.
43. Selker, T., Fischer, E. A., **Bederson, B. B.**, McCormack, C., Nass, C. (2003) Voting: User Experience, Technology and Practice. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2003) ACM Press, Panel, 700-701.
44. **Bederson, B. B.**, Clamage, A., Czerwinski, M. P., Robertson, G. G. (2003) A Fisheye Calendar Interface for PDAs: Providing Overviews in Small Displays. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2003) ACM Press, Demonstration, 618-619.
45. Hourcade, J.P., **Bederson, B. B.**, Druin, A., Rose, A., Farber, A., Takayama, Y. (2002). The International Children's Digital Library: Viewing Digital Books Online. Interaction Design and Children International Workshop, August 28-29, 2002, Eindhoven, The Netherlands 15 (3), 151-167.
46. Herrnson, P., **Bederson, B. B.** (2002) Debugging Maryland Balloting, Washington Post, Outlook Section, p. B08, Sunday May 12th, 2002.
47. Good, L., **Bederson, B. B.**, Stefik, M., Baudisch, P. (2002) Automatic Text Reduction For Changing Size Constraints. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2002) ACM Press, Short Paper, 798-799.

48. Hourcade, J.P., Bederson, B. B., Druin, A., Taxén, G. (2002) KidPad: A Collaborative Storytelling Tool for Children. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2002) ACM Press, Referred Demo, 500-501.
49. Hourcade, J.P., Bederson, B. B., Druin, A. (2002) SearchKids: A Digital Library Interface for Young Children. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 2002) ACM Press, Referred Demo, 512-513.
50. **Bederson, B. B.** (2001). PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps. UIST 2001, ACM Symposium on User Interface Software and Technology. Refereed demo.
51. Browne, H., Bederson, B. B., Druin, A., Sherman, L., & Westerman, W. (September 2000) Designing a Collaborative Finger Painting Application for Children, HCIL-2000-17, CS-TR-4184, UMIACS-TR-2000-66
52. Hourcade, J., Bederson, B. B. (May 1999) Architecture and Implementation of a Java Package for Multiple Input Devices (MID) HCIL-99-08, CS-TR-4018, UMIACS-TR-99-26
53. Hightower, R.R., Ring, L.T., Helfman, J.I., **Bederson, B. B.,** Hollan, J.D. (1998) PadPrints: Graphical Multiscale Web Histories. UIST 1998, ACM Symposium on User Interface Software and Technology. Tech Note, 121-122.
54. Stewart, J., Raybourn, E., Bederson, B. B., & Druin, A. (1998). When Two Hands Are Better Than One: Enhancing Collaboration Using Single Display Groupware. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 98) ACM Press, Short Paper, 287-288.
55. **Bederson, B. B.,** Hollan, J. D., Druin, A., Stewart, J., Rogers, D., & Proft, D. (1996). Local Tools: An Alternative to Tool Palettes. In Proceedings of User Interface and Software Technology (UIST 96) ACM Press, Short Paper, 169-170.
56. **Bederson, B. B.,** Hollan, J. D. (1995) Pad++: A Zoomable Graphical Interface System. In Proceedings of Conference on Human Factors in Computing Systems Conference Companion (CHI 95) ACM Press, Short Paper, 23-24.
57. **Bederson, B. B., &** Davenport, G., & Druin, A., & Maes, P. (1994). Immersive Environments: A Physical Approach to the Multimedia Experience, In Proceedings of IEEE International Conference on Multimedia Computing and Systems, Boston, Massachusetts, May 17-19, 1994, (Panel).
58. **Bederson, B. B.** (1995). Audio Augmented Reality: A Prototype Automated Tour Guide. In Proceedings of Human Factors in Computing Systems (CHI 95) New York: ACM Press, (Short Paper), 210-211.
59. Stone, L. M., Erickson, T., **Bederson, B. B.,** Rothman, P., Muzzy, R. (1994) Visualizing Data: Is Virtual Reality the Key? IEEE Visualization, IEEE Press, Short Paper, 410-413.
60. **Bederson, B. B.,** Stead, L., & Hollan, J. D. (1994). Pad++: Advances in Multiscale Interfaces. In Proceedings of Human Factors in Computing Systems (CHI 94) New York: ACM Press, Short Paper, 315-316.
61. **Bederson, B. B.** (1992). A Miniature Space-Variant Active Vision System: Cortex-I. Doctoral dissertation, New York University, New York, NY.

E. TALKS, ABSTRACTS, AND OTHER PROFESSIONAL PAPERS PRESENTED

i. Keynote Talks

1. April 2017 **CHIuXiD - The International Conference in Human-Computer Interaction & User Experience**, Jakarta, Indonesia
“From Human-Computer Interaction to Strategic Leadership for Change”
2. March 2017 **Maryland College Learning Center Association**, Shady Grove, MD
“Fishing for Faculty: Teaching & Learning Engagement at UMD”
3. April 2014 **Innovations in Teaching and Learning Conference**, College Park, MD
“An Introduction to the Teaching and Learning Transformation Center”

4. October, 2012 **South African Institute for Computer Scientists and Information Technologists**
Johannesburg, South Africa
"Social Participation: How Collective Activity Can Make Change"
5. September, 2010 **Visual Information Communication International Symposium**, Beijing, China
"The Promise of Zoomable User Interfaces"
6. May, 2010 **Human-Computer Interaction Lab's 27th Annual Symposium**, College Park, MD
"The Promise of Zoomable User Interfaces"
7. November, 2009 Raja Roy Singh Keynote Lecture at the **UNESCO Asia-Pacific Programme of Educational Innovation for Development**, Hangzhou, China
"Educational Technology for Creativity, Collaboration and Community"
8. February, 2009 **Microsoft Internal "UX Canvas Camp"**, Redmond, WA
"A Short History of Zoomable User Interfaces"
9. September, 2002 **IEEE Symposium on Human Centric Computing Languages and Environments (HCC 02)**, Arlington, VA
"Interfaces for Staying in the Flow"

ii. Invited Talks (doesn't include Keynote talks, which are listed above)

1. October 3, 2018, **Rutgers University**, Newark, NJ
"Transforming Teaching at the University of Maryland"
2. September 28, 2018, **EPAM Executive Roundtable**, Minsk, Belarus
"Workforce of the Future"
on Panel with Eugene Roman (Canada Tire), Sjoerd Gehring (Johnson & Johnson) & Sandra Loughlin (UMD)
3. September 27, 2018, **EPAM Software Engineering Conference (SEC 2018)**, Minsk, Belarus
"Open Company, Open Community, Open World"
on Panel with Elaina Shekhter (EPAM), Sjoerd Gehring (Johnson & Johnson), Scott Nissenbaum (Ben Franklin Technology Partners) & Sandra Loughlin (UMD)
4. December 4, 2017, **Cornell Tech**, New York, NY
"Academic Technologies in Higher Ed"
5. November 10, 2017, **Online Programs and MOOCs – Opportunities and Challenges**, in The 7th Annual MIS Academic Leadership Conference, RHTSmith School of Business, University of Maryland
6. Sept 2, 2017, **Google**, New York, NY
"Academic Technologies in Higher Ed"
7. June 11, 2015 **Designing a Learning Environment**, in "Open Textbooks: Challenges for Policy" workshop, University of Haifa, Haifa, Israel
8. May 13, 2015 **Strategic Integration of MOOCs into Graduate and Professional STEM Programs**, Champaign, Illinois
Raineri, D., Breslow, L., **Bederson, B. B.**
"Future of MOOCs in Research Universities"
9. April 13, 2015 **#EduAnalyticsDC Workshop**, Washington, DC
"Understanding Consistency in a Large Multi-Section Course with Many Instructors"
10. March 3, 2015 **Coursera Partners Conference**, Newport Beach, CA
Verstelle, M., Bennett, A., **Bederson, B. B.**, & Jeffries, Pam
"Designing a MOOC Portfolio: Content Strategy for Online Education"
11. March 1, 2015 **Coursera Partners Conference**, Newport Beach, CA
Buitendijk, S., **Bederson, B. B.**, Kerrison, M., & Price, V.
"Value to Partners: Case Studies From Partner Institutions"

12. January 29, 2015 **University of Wisconsin, Madison**, Madison, WI
 “Reflections on 20 Years Doing HCI Systems Research”
 Department of Computer Science, Distinguished Lecture Series
13. Sheil, M., Rock, E., O’Brien, L., **Bederson, B. B.**, & Xiaoming, L. (2014) “Future Trajectories of MOOCs in Higher Ed”, Panel at Coursera Partners’ Conference.
14. February 12, 2014 **Patuxent Partnership**, Lexington Park, MD
 “UMD MOOCs and What the Future Holds For Them”
15. January 29, 2014 **Carnegie Mellon University**, Pittsburgh, PA
 “Power to the People: Approaches to Crowdsourcing”
16. January 14, 2014 **Microsoft**, Friendship Heights, MD
 “Cracking the Code: Improving Computer Science Education Outcomes in the State of Maryland”
 Panel participant on Microsoft public series “Conversations on Education”
17. November 20, 2013 **University of Maryland**, College Park, MD
 “Mobile, Online, Flipped, Active: New Uses of Educational Technologies”
 College of Education
18. October 11, 2013 **Huawei**, Hong Kong, China
 “When Crowds Come Together”
19. October 8, 2013 **National Institute of Information and Communications Technology**, Osaka, Japan
 “Crowdsourced Monolingual Translation”
 in 2nd Workshop on Future Directions in Translation Research
20. February 1, 2013 **Google**, Mountain View, CA
 “Social + MOOCs + Hcomp = SMOOCH: Computing Comes to Education”
21. January 29, 2013 **Google**, Mountain View, CA
 “The Promise of Zoomable User Interfaces: Reflections on Design of Simple UX”
22. December 6, 2012 **Sunlight Foundation**, Washington, DC
 “Digital Disclosure of a Nationally Standardized Database of Restaurant Food Safety Inspections” (with Ginger Jin)
23. November 19, 2012 **University of Maryland**, College Park, MD
 “Innovation, Iteration & Creation: A Discussion about the Development of New Ideas in the Modern World”
24. November 6, 2012 **World Usability Day**, Minsk, Belarus
 “The Promise of Zoomable User Interfaces”
25. November 2, 2012 **Central & Eastern European Software Engineering Conference in Russia** Moscow, Russia
 “Learning to Code in an Online World”
26. November 2, 2012 **Central & Eastern European Software Engineering Conference in Russia** Moscow, Russia
 Panelist in “Peopleware: what skills and qualifications we need to build a dream team of software engineers”
27. October 4, 2012 **UNISA – University of South Africa**, Johannesburg, South Africa
 “Innovation, Iteration & Creation: A Discussion about the Development of New Ideas in the Modern World”
28. June 20, 2012 **University of Haifa**, Haifa, Israel
 “Books that are Blue: What Children Can Teach Us About Creating New Technologies” (with Allison Druin)
29. June 19, 2012 **Interdisciplinary Center**, Herzilya, Israel
 “Books that are Blue: What Children Can Teach Us About Creating New Technologies” (with Allison Druin)

30. Nov 17, 2011 **Future of Information Alliance**, College Park, MD
“Anonymity and Privacy While Learning”
31. Sept 21, 2011 **U.S. Senate and Congress Briefings**, Washington, DC
Panel participant in “Deconstructing the iPad: How Federally Supported Research Leads to Game-Changing Innovation” run by the Task Force on American Innovation and the CRA
32. Aug 24, 2011 **Summer Social Webshop** College Park, MD
“Translation by Collaboration Among Monolingual Users”
33. May, 2011 **National Science Foundation**, Arlington, VA
Distinguished Lecture Series
“Translation by Collaboration among Monolingual Users” (with Philip Resnik)
34. September, 2010 **Microsoft Research Asia**, Beijing, China
“Collaborative Monolingual Translation”
35. September, 2010 **Tsinghua University**, Beijing, China
“Collaborative Monolingual Translation”
36. April, 2010 **CHI 2010**, Atlanta, GA
Social Impact Award talk
37. October, 2009 **Google**, Mountain View, CA
Panel participant in “Breakthrough Learning in a Digital Age” invitation only conference
38. September, 2009 **Google Tech Talk**, Mountain View, CA
“Translation by Iterative Collaboration between Monolingual Users”
39. April, 2009 **University of Illinois (CS Dept.)**, Champaign-Urbana, IL
“Zoomable User Interfaces from Sun Sparcstations to iPhones”
40. April, 2009 **University of Illinois (iSchool)**, Champaign-Urbana, IL
“From Ulaan Baatar to iPhone: A Digital Library for the World’s Children”
41. February, 2009 **Tufts University**, Medford, MA
“From iPhone to Ulaan Baatar: A Digital Library for the World’s Children”
42. February, 2009 **Microsoft Internal “UX Canvas Camp”**, Redmond, WA
“A Short History of Zoomable User Interfaces”
43. September, 2008, **Carnegie Mellon University**, Pittsburgh, PA
“Black Ears to Blonde Cats: Paths for Designing Change”
44. May, 2008, **Manchester Public Library** Manchester by the Sea, MA
“No Road, Drive: The ICDL Travels to Mongolia”
45. April, 2008 **Web 2.0 Expo**, San Francisco, CA
“Next Generation Mobile UI” (with John SanGiovanni)
46. February, 2008 **USACM Executive Council**, Washington, DC
“Voting Technology: The Not-So-Simple Act of Casting a Ballot”
47. October, 2007 **Widget Summit**, San Francisco, CA
“Mobile Widget Design”
48. September, 2007 **Knight Center for Specialized Journalism**, College Park, MD
“Mobile Technology Trends”
49. June, 2007 **Mobile Marketing Association**, New York, NY
“Designing for Usability”
50. May, 2007 **MIT Enterprise Forum**, Arlington, VA
“Co-present Mobile Collaboration”
51. December, 2006 **Human Factors and Ergonomics Society Potomac Chapter**, Bethesda, MD
“A Library for the World’s Children”

52. June, 2006 **Open Society Forum**, Ulaan Baatar, Mongolia
"A Library for the World's Children"
53. February, 2006 **Dingman Center for Entrepreneurship, R.H. Smith School of Business, UMD**
"Web 2.0 UI"
54. December, 2005 **World Bank**, Washington, DC
"The International Children's Digital Library"
55. December, 2005 **IBM T.J. Watson Research Center**, Hawthorne, NY
"Making Visualization Work"
56. November, 2005 **NSA**, Laurel, MD
"Making Visualization Work"
57. October, 2005 **IBM Research**, Cambridge, MA
"Making Visualization Work"
58. October, 2005 **Johns Hopkins University – Computer Science Dept.**, Baltimore, MD
"Making Visualization Work"
59. September, 2005 **InfoToday Conference on 'Innovations in Search'**, New York, NY
"Zooming Into Visual Search"
60. March, 2005 **NFAIS**, Philadelphia, PA
"Making Sense of Search Results"
61. February, 2005 **American Association for the Advancement of Science (AAAS)** Washington, DC
Panel in annual conference,
"Designing Interfaces for Voting Machines"
62. February, 2005 **National Association of State Election Directors (NASED)** Washington, DC
Invited talk at annual conference.
"Designing Interfaces for Voting Machines"
63. January, 2005 **Microsoft**, Seattle, WA
"Interfaces for Staying in the Flow"
64. January, 2005 **Google**, Mountain View, CA
"The International Children's Digital Library"
65. December, 2004 **National Library of New Zealand**, Wellington, New Zealand
"The International Children's Digital Library"
66. December, 2004 **University of Waikato**, Hamilton, New Zealand
"Interfaces for Staying in the Flow"
67. December, 2004 **Victoria University Wellington**, Wellington, New Zealand
"Interfaces for Staying in the Flow"
68. December, 2004 **University of Canterbury**, Christchurch, New Zealand
"Interfaces for Staying in the Flow"
69. November, 2004 **National Archives and Records Administration (NARA)**, College Park, MD
in Conference "Partnerships in Innovation: Serving a Networked Nation"
"International Children's Digital Library"
70. April, 2004 **Johns Hopkins University Applied Physics Lab**, Laurel, MD
"Interfaces for Staying in the Flow"
71. October, 2003 **University of Maryland, Robert H. Smith School of Business**, College Park, MD
on InForum 2003 panel on "Does IT Matter?"
72. September, 2003 **University of Maryland**, College Park, MD
"Making Visualization Work"
73. September, 2003 **University of Victoria**, Victoria, Canada
"Interfaces for Staying in the Flow"

74. March, 2003 **National Archives and Records Administration**, College Park, MD
"Visualizations for Archivists"
75. March, 2003 **Microsoft Research**, Redmond, WA
"DateLens: A Scalable Fisheye Calendar Interface"
76. January, 2003 **Sun Microsystems**, Menlo Park, CA
"Zoomable User Interfaces for Searching and Browsing"
77. September, 2002 **Microsoft Research Limited**, Cambridge, England
"Information Visualization for Mobile Devices"
78. July, 2002 **Microsoft Research**, Redmond, WA
Demo of DateLens with Bill Gates at Microsoft Faculty Summit
79. July, 2002 **Microsoft Research**, Redmond, WA
"Scalable Interfaces for Mobile Devices"
80. June, 2002, **WebShop**, Dept. of Sociology, UMD, College Park, MD
"Access and Usability: The Importance of Internet Applications"
81. January, 2002 **Workshop on Election Standards and Technology**, Washington, DC
"Visualization Techniques for Electronic Voting"
82. October, 2001 **Microsoft Research**, Redmond, WA
"Zoomable User Interfaces for Searching and Browsing: SearchKids & PhotoMesa"
83. October, 2001 **Xerox PARC**, Palo Alto, CA
"Zoomable User Interfaces for Searching and Browsing: SearchKids & PhotoMesa"
84. November, 2000 **Census Bureau, Dept. of Commerce**, Suitland, MD
"Zoomable User Interfaces"
85. May, 2000 **Bowie State University**, Bowie, MD
"Zoomable User Interfaces and Single Display Groupware"
86. April, 2000 **University of Delaware**, DE
"Zoomable User Interfaces and Single Display Groupware"
87. August, 1999 **Xerox PARC**, Palo Alto, CA
"Zoomable User Interfaces and Single Display Groupware"
88. August, 1999 **University of California, Berkeley**, Berkeley, CA
"Zoomable User Interfaces and Single Display Groupware"
89. August, 1999 **Sun Microsystems, Inc. (Java Group)**, Palo Alto, CA
"Zoomable User Interfaces and Single Display Groupware"
90. July, 1999 **Nokia**, Linköping Sweden
"Zoomable Information Retrieval Systems and Single Display Groupware"
91. June, 1998 **IBM**, Yorktown, NY
"Zoomable User Interfaces"
92. May, 1998 **Washington DC Linux User's Group**, Alexandria, VA
"Zoomable User Interfaces"
93. April, 1998 **National Institute of Standards and Technology (NIST)**, Gaithersburg, MD
"Zoomable User Interfaces"
94. March, 1998 **Microsoft Research**, Redmond, WA
"Zoomable User Interfaces"
95. October, 1997 **Netherlands Design Institute**, Amsterdam, Holland
"Pad++: Zoomable User Interfaces"
96. October, 1997 **Philips Labs**, Eindhoven, Holland
"Pad++: Zoomable User Interfaces"

97. October, 1997 **CID, Royal Institute of Technology (KTY)**, Stockholm, Sweden
"Pad++: Zoomable User Interfaces"
98. September, 1997 **Swedish Institute of Computer Science (SICS)**, Stockholm, Sweden
"Pad++: Zoomable User Interfaces"
99. October, 1996 **ART3000, A French society for the development of Art & Technology**, Paris, France
"Pad++: A Zoomable Graphical Interface"
100. October, 1996 **University of Michigan, Department of Information Sciences**, Ann Arbor, MI
"Pad++: A Zoomable Graphical Interface"
101. July, 1996 **AT&T Bell Labs**, NJ
"Pad++: A Zoomable Graphical Interface"
102. January, 1996 **Stanford University, Cognitive Psychology Group**, CA
"Pad++: A Zoomable Graphical Interface"
103. October, 1995 **Cornell University, Computer Science Department**, Ithaca, NY
"Pad++: A Zoomable Graphical Interface"
104. June, 1995 **Sandia National Labs**, Los Alamos, NM
"Pad++: A Zoomable Graphical Interface"
105. April, 1995 **Los Alamos National Labs**, Los Alamos, NM
"Pad++: A Zoomable Graphical Interface"
106. November, 1993 **Digital Equipment Corporation, Stanford Research Center**, CA
"Pad++: A Multiscale Hierarchical Sketchpad"
107. October, 1993 **Brown University, Computer Science Department**, Providence, RI
"Pad++: A Multiscale Hierarchical Sketchpad"
108. April, 1993 **University of Chicago, Computer Science Department**, |
"PA3D: A Multiscale Hierarchical Sketchpad"
109. October, 1992 **Rensselaer Polytechnic Institute, Computer Science Department**, Troy, NY
"Cortex-I: A Miniature Active Vision System"

H. ORIGINAL DESIGNS, PLANS, INVENTIONS, AND PATENTS

i. Patents

1. SanGiovanni, J., & **Bederson, B. B.** (Inventors). (2022) *Systems, methods, and computer program products displaying interactive elements on a canvas*. US Patent #11,243,673
2. **Bederson, B. B.**, & SanGiovanni, J. (Inventors). (2021) *Techniques to modify content and view content on a mobile device*. US Patent #11,061,524.
3. SanGiovanni, J., & **Bederson, B. B.**, and Painter, B. (Inventors). (2018) *Systems and methods for adaptive third party content*. US Patent #10,019,963.
4. SanGiovanni, J., & **Bederson, B. B.** (Inventors). (2018) *Systems, methods, and computer program products displaying interactive elements on a canvas*. US Patent #9,959,020.
5. **Bederson, B. B.**, & SanGiovanni, J. (Inventors). (2017) *Techniques to modify content and view content on mobile devices*. US Patent #9,778,810.
6. SanGiovanni, J., & **Bederson, B. B.** (Inventors). (2016) *Systems, Methods, and Computer Program Products Providing an Integrated User Interface for Reading Content*. US Patent #9,361,130.
7. SanGiovanni, J., & **Bederson, B. B.** (Inventors). (2014) *Systems, Methods, and Computer Program Products Displaying Interactive Elements on a Canvas*. US Patent #8,819,570.
8. Pahud, M., Murillo, O. E., Karlson, A. K., & **Bederson, B. B.** (Inventors). (2012). *Monitoring Pointer Trajectory and Modifying Display Interface*. US Patent #8,261,211.

9. Good, L.E., **Bederson, B. B.**, & Stefik, M.J. (Inventors). (2010). *Methods and Systems for Supporting Presentation Tools Using Zoomable User Interfaces*. US Patent #7,707,503.
10. **Bederson, B. B.**, Good, L. E., & Stefik, M.J. (Inventors). (2009). *Methods and Systems for Incrementally Changing Text Representation*. US Patent #7,650,562 B2.
11. **Bederson, B. B.**, Good, L. E., & Stefik, M. J. (Inventors). (2009). *Methods and Systems for Incrementally Changing Text Representation*. US Patent #7,549,114 B2.
12. Wallace, R. S., **Bederson, B. B.**, & Schwartz, E. L. (Inventors). (1997). *TV Picture Compression and Expansion*. US Patent #5,642,167.
13. **Bederson, B. B.**, Wallace, R. S., & Schwartz, E. L. (Inventors). (1993). *Two-Dimensional Pointing Motor*. US Patent #5,204,573.
14. Wallace, R. S., **Bederson, B. B.**, & Schwartz, E. L. (Inventors). (1992). *Telephone Line Picture Transmission*. US Patent #5,175,617.

ii. University Disclosures

1. **Bederson, B. B.** & Lin, J. (Inventors) (2009), CrowdFlow: A Computer-Human Hybrid Cloud Computing Model.
2. **Bederson, B. B.** (Inventor) (2002), Fisheye Calendar: A Calendar interface for small to large devices, IS-2002-015.
3. **Bederson, B. B.** (Inventor) (2001), PhotoMesa: A Zoomable Image Browser, IS-2001-053
4. **Bederson, B. B.**, & Chipman, L. G. (Inventors) (2000), *Physical Linear Slider Device to Control Computer Interface*, IS-2000-018.
5. **Bederson, B. B.** (Inventor) *Fisheye Menus For Selecting an Item From a Long List*, IS-2000-025.
6. **Bederson, B. B.**, Druin, A., & Pablo, J. P. (Inventors) (2000), *KidPad: A Collaborative Storytelling Environment for Children*, IS-2000-019.

I. CONTRACTS AND GRANTS

Total Funding: \$18,036,144

UMD Portion: \$10,329,299

1. **Ithaka SR, July 2016, co-PI**
Elementary Statistics with an adaptive learning system
\$113,500
2. **Maryland Innovation Initiative, March 2014, PI**
A Nationwide Database of Food Safety Inspections for Retail Establishments
\$100,000
3. **Huawei, July 2013, PI**
XParty: Synchronous Online Language Learning
\$54,000
4. **Google, March 2013, PI**
XParty: Synchronous Peer Learning in MOOCs (extension)
\$48,872
5. **Google, December 2011, PI**
Search Party: Learning to Search in a Web-Based Classroom
\$60,000
6. **Alfred P. Sloan Foundation, November 2011, co-PI**
Digital Disclosure of a Nationally Standardized Database of Restaurant Food Safety Inspections
\$465,272

7. **Google, June 2011, PI**
Search Party: Web-Based Classroom Learning to Search
\$15,000
8. **Google, June 2011, co-PI**
Translate the World: A Unified Framework for Crowdsourcing Translation
\$150,000
9. **Nokia Research, July 2010, PI**
Mobile Learning
\$36,000
10. **NSF, CDI Type I: Translation as a Collaborative Process, Sep 2009, PI**
\$630,000
11. **Google, July 2009, PI**
Translation as a Collaborative Process
\$70,000
12. **Sesame Workshop, January 2009, co-PI**
Literacy 360 Alliance
\$22,500
13. **Eliassen Foundation, Nov 2008, co-PI**
ICDL Elias Project
\$50,000
14. **Intel, Nov 2008, co-PI**
ICDL ClassmatePC Project
\$50,000
15. **NSF, Designing and Understanding Intergenerational Mobile Learning Communities, Sep 2008, co-PI**
\$100,000
16. **Government of Mongolia Ministry of Education, Culture and Science – Nov2008, PI**
International Children’s Digital Library Phase 2
\$230,000
\$190,000 (for UMD)
17. **NSF, Usability Study of Independent Voter Verification Systems – July 2006, co-PI**
PI – Paul Herrnson (Gov’t & Politics)
\$68,000
18. **Government of Mongolia Ministry of Education, Culture and Science – May 2006, PI**
Creating Digital Libraries in Mongolia
\$75,000
19. **IBM Faculty Award – December 2005, PI**
Award to support my research on the open source Piccolo graphics toolkit.
\$30,000
20. **Maryland State Board of Elections, September 2005 – January 2006, co-PI**
PI – Paul Herrnson (Gov’t & Politics)
\$70,000
21. **(Agency not disclosed), April 2005 – March 2006, co-PI**
Joint Institute for Knowledge Discovery
\$224,000 (my part of larger \$2,000,000 effort led by VS Subrahmanian)
22. **Booz Allen (subcontract from ARDA), March 2005 – April 2006**
Human-Information Interaction Research
\$325,000

23. **Toshiba, July 2004 – December 2006, PI**
Automatic Thumbnail Cropping
\$20,000
24. **Microsoft, January 2004 – December 2006, PI**
Interaction Design and Visualization
\$1,000,000
25. **DARPA, February, 2003 – January 2006, PI**
Zoomable User Interfaces for the Semantic Web
\$230,000
26. **NSF, September, 2003 – August 2006, co-PI**
PI – Paul Herrnson (Gov't & Politics), co-PI, Richard Niemi (Univ. of Rochester), Michael Traugott (Univ. of Michigan)
An Assessment of Voting Technology and Ballot Design
\$900,000 – approx \$540K for UMD research
27. **Chevron, Oct 2002 – Oct 2003, co-PI**
PI – Catherine Plaisant (UMIACS)
Information Visualization for the Oilfield of the Future
\$75,000
28. **IBM, September, 2002, PI**
T221 high-resolution display
\$9,000 in kind
29. **IMLS, September, 2002 – August, 2005, co-PI**
PI – Allison Druin (CLIS)
The Children's International Digital Library
\$397,162
30. **NSF, September, 2002 – August 2005, PI**
Search Interfaces for Biodiversity Informatics
\$380,000
31. **NSF, September 2002 – August 2007, co-PI**
PI – Allison Druin (CLIS), co-PI, Anne Weeks
Developing a Children's International Digital Library
\$3,000,000 (\$2,125,000 to UMD)
32. **NSF, September 2002 – August 2005, co-PI**
co-PI – Allison Druin (CLIS), PI - Nancy Kaplan (Univ. of Baltimore), Stuart Malthrop (U. of Balt.)
An HCI Partnership Serving Under represented Groups
\$420,000 (\$133,000 to UMD)
33. **Microsoft Research, May 2002 – April 2001, PI**
Mobile Interfaces and .NET
\$100,000
34. **Microsoft Research, September 2001 – August 2002, PI**
Mobile Interfaces
\$50,000
35. **DARPA, May 2001 – August 2003, PI**
Zoomable User Interfaces for the Semantic Web
\$320,000
36. **European Union – Esprit, The Disappearing Computer, Jan 2001 – December 2003, co-PI**
co-PI – Allison Druin (CLIS), Several PIs in Europe
With Royal Institute of Technology (KTH – Sweden), Univ. of Paris Sud, France.
InterLiving: Designing Interactive, Intergenerational Interfaces for Living Together
~\$1,500,000 - approx \$400K for UMD research

37. **Chevron, Nov 2000 – May, 2002, PI**
co-PI – Catherine Plaisant (UMIACS)
 Information Visualization
 \$141,000
38. **DARPA, May 1998 – June 2002, PI**
 Zoomable User Interfaces for InImEx and Command Post of the Future (CPOF)
 \$1,000,000
39. **Lockheed Martin Corp. sub-contract from DARPA, June 1999 - June 2000, PI**
 Unified Zoomable Environment (UZE) for Command Post of the Future
 \$40,000
40. **European Union – Esprit, Intelligent Information Interfaces, Sept. 1998 – August 2001, co-PI**
co-PI – Allison Druin (CLIS), Several PIs in Europe
 With Swedish Institute of Computer Science, Royal Institute of Technology (KTH – Sweden)
 University of Nottingham, England.
 KidStory
 ~\$1,500,000 - approx \$300K for UMD research
41. **Sun Microsystems, May 1999, PI**
 Zoomable User Interfaces
 \$30,000 in equipment
42. **UNM sub-contract from DARPA, February 1998 – March 1999, PI**
 Pad++: Zoomable User Interfaces
 \$200,000
43. **Microsoft, April, 1998, PI**
 Productivity and development software
 \$5,000 in software and tech support
44. **Trivista, PI January 1998 - January 1999, PI**
 Graphical User Interfaces based on Zooming
 \$50,000

UNIVERSITY OF NEW MEXICO

45. **Sony Research Corporation, PI**
co-PI – Jim Hollan (Univ. of New Mexico), co-PI – Ken Perlin (New York University)
 Licensing of Pad++ software from UNM and NYU
 \$500,000 plus royalties
46. **Intel Donation, January, 1997, co-PI**
PI – Jim Hollan (Univ. of New Mexico)
 200 MHz PC Computer
 \$4,000 in-kind
47. **DARPA, July '94 - June '97, Worked on grant with Jim Hollan**
 Beyond Imitation: A Strategy For Building A New Generation of HCI Design Environments
 \$3,000,000 over 3 years with subcontracts to New York University and the Univ. of Michigan
48. **Sandia LDRD (Sandia internal research project) Oct'95 - Oct'96, PI**
 Data Zooming - A New Physics for Information Navigation
 \$64,000
49. **Sandia SURP (Sandia University Research Program) Oct'95 - Oct'96, PI**
 Pad++: A Zoomable Interface to the World-Wide Web
 \$40,000
50. **Sony Research Corporation, Jan'96, PI**
co-PI – Allison Druin (University of New Mexico)
 Help plan Sony's efforts to develop a Zooming User Interface system
 \$3,674

51. **Sandia LDRD (Sandia internal research project) Oct'94 - Oct'95, Worked on grant with Jim Hollan**
\$40,000
52. **Bellcore Donation, July'93, PI**
SGI Indigo Workstation computer
\$30,000 in-kind

J. FELLOWSHIPS, PRIZES, AWARDS

1. **2022 – IEEE VIS Test of Time Award.** Joint with Catherine Plaisant and Jesse Grosjean – for my 2002 paper: SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation. *IEEE Symposium on Information Visualization, 2002. INFOVIS 2002.*, 2002, pp. 57-64, doi: 10.1109/INFVIS.2002.1173148. <https://ieevis.org/year/2022/info/awards/test-of-time-awards>
2. **2021 – IEEE VIS Test of Time Award.** Joint with Ben Shneiderman and Martin Wattenberg – for my 2001 and 2002 papers: Ordered and Quantum Treemaps: Making effective use of 2D space to display hierarchies. Benjamin B. Bederson, Ben Shneiderman, Martin Wattenberg. *ACM TOG* 21(4):833-854 200
DOI: 10.1145/571647.571649. <https://ieevis.org/year/2021/info/awards/test-of-time-awards>
3. **2016 – Winner of Best Historical Paper Award.** Joint with Pekka Parhi and Amy K. Karlson
MobileHCI 2016 – 18th International Conference on Human-Computer Interaction With Mobile Devices and Services for “Target Size Study for One-Handed Thumb Use on Small Touchscreen Devices”, MobileHCI 2006.
4. **2012 – Elected to the CHI Academy**
<http://www.sigchi.org/about/awards/>

The CHI Academy is an honorary group of individuals who have made substantial contributions to the field of human-computer interaction. These are the principal leaders of the field, whose efforts have shaped the disciplines and/or industry, and led the research and/or innovation in human-computer interaction. The criteria for election to the CHI Academy are:

- Cumulative contributions to the field.
- Impact on the field through development of new research directions and/or innovations.
- Influence on the work of others.

5. **2011 – Recognized as an ACM Distinguished Scientist**
http://awards.acm.org/distinguished_member/year.cfm

The Distinguished Member Grade recognizes those ACM members with at least 15 years of professional experience and 5 years of continuous Professional Membership who have achieved significant accomplishments or have made a significant impact on the computing field.

6. **2010 – ICDL received the American Library Association President’s Award for International Library Innovation.** Joint with Allison Druin and Ann Weeks.
7. **2010 – Winner of SIGCHI Social Impact Award**
<http://www.sigchi.org/about/awards/2010-sigchi-awards>
8. **2009 – Winner of Digital Education Achievement Award from the Center for Digital Education and Converge magazine**
For joint work on the International Children’s Digital Library with Allison Druin and Ann Weeks.
9. **2007 – Winner of Brian Shackel Award for “outstanding contribution with international impact in the field of HCI”**
Karlson, A., & Bederson, B. B. (2007) ThumbSpace: Generalized One-Handed Input for Touchscreen-Based Mobile Devices, *Proceedings of INTERACT 2007*, 324-338.
10. **2005 – IBM Faculty Award**
\$30,000 gift to support my work on open source Piccolo graphics toolkit

11. **2005 – The International Children’s Digital Library** (www.icdlbooks.org) website was selected for inclusion by the National Endowment for the Humanities on their EDSITEment website (<http://edsitement.neh.gov>) as one of the best online resources for education in the humanities.
12. **2004 – Winner of InfoVis Contest**
Lee, B., Czerwinski, M., Robertson, G. G. &, Bederson, B. B. (2004). “Understanding Eight Years of InfoVis Conferences using PaperLens”, InfoVis 2004, IEEE, 216-217.
13. **2003 – Winner of Best Student Paper Award**
Suh, B., Ling, H., Bederson, B. B., & Jacobs, J. W. (2003). “Automatic Thumbnail Cropping and Its Effectiveness”. UIST 2003, ACM Symposium on User Interface Software and Technology, CHI Letters, 5(2), 95-104.
14. **1992 – Janet Fabri Memorial Award for Outstanding Doctoral Dissertation**,
Department of Computer Science, Courant Institute, New York University

K. REVIEWING ACTIVITIES

ACM Conference on Graphics (SIGGRAPH)
ACM Conference on Human Factors in Computing Systems (CHI)
ACM Conference on Information Retrieval (IR)
ACM Conference on User Interfaces and Software Technology (UIST)
ACM Transactions on Computer-Human Interaction (TOCHI)
Communications of the ACM (CACM)
Computer Graphics and Applications
Graphics Interface Conference, Canada (GI)
International Conference on Information Visualization
International Journal of Human-Computer Interaction
IEEE Symposium on Information Visualization (InfoVis)
IEEE Symposium on Visual Analytics Science and Technology (VAST)
IEEE Transactions on Software Engineering (TSE)
IEEE Transactions on Visualization and Computer Graphics (TVCG)
Journal of Experimental Algorithmics
Journal of Information Visualization
Journal of Software Practice and Engineering (SPE)
NSERC (Canadian scientific funding agency)
National Science Foundation

L. OTHER

i. Software

StoryKit	iPhone app for children and their elders to create new books. Put in iPhone App Store August, 2009. Typical usage of 100,000 hours of story writing per month.
ICDL for iPhone	iPhone app for reading books from the International Children's Digital Library. Put in iPhone App Store November, 2008.
International Children's Digital Library	A library of children's books. Put on the Web Nov, 2002. http://www.childrensbooks.org • Unique accessors total: > 7,000,000 • Unique accessors monthly: ~130,000
DateLens	A calendar program for PDAs. Put on the Web July, 2002
SpaceTree	A general purpose tree browser. Put on the Web Dec, 2002. http://www.cs.umd.edu/hcil/spacetree
PhotoMesa	An end-user zoomable photo browser. Put on the Web May, 2001. http://www.cs.umd.edu/hcil/photomesa
CounterPoint	A zoomable presentation tool that works as a plug-in for Microsoft PowerPoint. Put on the Web in late 2001. http://www.cs.umd.edu/hcil/counterpoint
Piccolo	A library to support structured 2D graphics, the successor to jazz. Put on the Web in July 2002. http://www.cs.umd.edu/hcil/piccolo
Jazz	A general-purpose Java toolkit for building Zoomable User Interfaces. Put on the Web as Open Source software July 1999. http://www.cs.umd.edu/hcil/jazz
KidPad	An application that teaches young children communication skills through collaborative storytelling. Built using MID and KidPad. Put on the Web for non-commercial use March 2000. Access through March 2001 follow. http://www.cs.umd.edu/hcil/kidpad
Fisheye Menus	A new user interface widget for selecting one item for a long list using graphical distortion. Put on the Web for non-commercial use June 2000. Access through March 2001 follow. http://www.cs.umd.edu/hcil/fisheyemenu
MID	A general-purpose Java toolkit for supporting multiple input devices. It currently supports multiple USB mice on Windows 98. Put on the Web as Open Source software July 1999. It is used by the small but growing community of researchers investigating Single Display Groupware http://www.cs.umd.edu/hcil/mid
Pad++	Pad++ is described in detail in the papers referenced above. http://www.cs.umd.edu/hcil/pad++
Togl	An extension to Tcl/Tk that allows the creation of OpenGL graphics written in C to appear in a window managed by Tk. In addition, Tcl event bindings can be attached to the window which can then communicate with the C code that did the OpenGL rendering. Put on the net for public access. http://togl.sourceforge.net/

ii. Media

1. February 2, 2017 **The Diamondback** by Carly Taylor
"McKeldin Mall's Newest Academic Building Will Be Completed in March After Nearly 3 Years"
<http://www.dbknews.com/2017/02/02/edward-st-john-construction-umd/>

2. August 29, 2016 **The Diamondback** by Andrew Dunn
 “University System of Maryland to offer an online course for free”
<http://www.dbknews.com/2016/08/30/mooc-umb-umuc-online-usm/>
3. February 15, 2016 **The Diamondback** by Samantha Reilly
 “SGA passes resolution to revive 2012 syllabus bill”
<http://www.dbknews.com/2016/02/16/umd-sga-passes-syllabus-resolution-to-revive-university-senate-bill/>
4. September 24, 2015 **The Diamondback** by Taylor Swaak
 “Active-Learning Courses Gain Traction at the University of Maryland”
http://www.diamondbackonline.com/news/active-learning-courses-gain-traction-at-the-university-of-maryland/article_435098ca-6247-11e5-a8e2-df59d97181a5.html
5. September 14, 2015 **Washington Post Express** by Beth Luberecki
 “Massive open online courses (MOOCs) can help you achieve the next step in your career”
<https://www.washingtonpost.com/express/wp/2015/09/14/massive-open-online-courses-moocs-can-help-you-achieve-the-next-step-in-your-career/>
6. April 30, 2015 **The Diamondback** by Talia Richman
 “USM Center for Academic Innovation to be renamed to honor Brit Kirwan”
http://www.diamondbackonline.com/news/article_51cbc02c-eeb7-11e4-a838-6b9857ba8e37.html
7. April 13, 2015 **The Diamondback** by Rokia Hassanein
 “UMD Team Creates Nation’s Largest Food Safety Inspection Database”
http://www.diamondbackonline.com/news/article_553da506-e181-11e4-b6c9-0f23d720b7cc.html#.VS1VV_HbNqc.twitter
8. April 6, 2015 **Quality Assurance & Food Safety Magazine**
 “Database of Food Safety Inspection Information Created”
<http://www.qualityassurancemag.com/food-safety-inspection-information.aspx>
9. February 23, 2015 **Diamondback, UMD Student Newspaper** by Taylor Swaak
 “New UMD website offers alternative ways to save money on textbooks”
http://www.diamondbackonline.com/news/article_0d910a34-baf5-11e4-826a-dfbc9e8b5f8.html
10. January 13, 2015 **WAMU Kojo Nnamdi Show**
 “The Rise of Mobile Apps”
http://thekojoonnamdishow.org/shows/2015-01-13/tt_the_rise_of_mobile_apps
11. October 1, 2014 **The Chronicle of Higher Education**
 “Optimism About MOOCs Fades in Campus IT Offices”
<http://chronicle.com/blogs/wiredcampus/optimism-about-moocs-fades-in-campus-it-offices-survey-finds/54705>
12. September 8, 2014 **WAMU Kojo Nnamdi Show**
 “Beefing Up Photo, Cloud and Online Security”
<http://thekojoonnamdishow.org/shows/2014-09-09/beefing-photo-cloud-and-online-security>
13. April 29, 2014 **Diamondback, UMD Student Newspaper** by James Levin
 “Learning to ask for help”
http://www.diamondbackonline.com/news/article_632cbbb8-cf5b-11e3-b8b4-0017a43b2370.html
14. April 18, 2014 **CNN.com** by Emanuella Grinberg
 “Removing ‘barriers’ to education through free college textbooks”
<http://www.cnn.com/2014/04/18/living/open-textbooks-online-education-resources/>
15. January 27, 2014 **DiamondBack, UMD Student Newspaper** by Ellie Silverman
 “University expanding online class sequences through Vanderbilt partnership”
http://www.diamondbackonline.com/news/campus/article_dff739f4-86ed-11e3-9a18-001a4bcf6878.html
16. September 27, 2013 **Baltimore Business Journal** by Sarah Gantz
 “Maryland Schools use Online Model to Tout Brand, Attract Students”
<http://www.bizjournals.com/baltimore/print-edition/2013/09/27/maryland-schools-use-online-model-to.html?page=all>

17. July 16, 2013 **HuffingtonPost.com**
“Blended Learning: College Classrooms of the Future”
http://www.huffingtonpost.com/uloop/blended-learning-college-_b_3598718.html
18. December 5, 2012 **Diamondback, UMD Student Newspaper** by Allison Gray
“University members hoping to see ‘massive open online courses’ integrated”
http://www.diamondbackonline.com/news/article_4873330c-3ef1-11e2-9a16-0019bb30f31a.html
19. September 21, 2012 **SmartMoney.com** by Quentin Fottrell
In Apple’s Maps War vs. Google, Consumers Lose
http://blogs.smartmoney.com/advice/2012/09/21/in-apples-maps-war-vs-google-consumers-lose/?link=SM_hp_ls4e
20. September 18, 2012 **WAMU Kojo Nnamdi Show**
“Creativity and the Patent Wars” (guest on one-hour show)
<http://thekojoannamdishow.org/shows/2012-09-18/creativity-and-patent-wars>
21. September 7, 2012 **New York Magazine** by Matthew Shaer
“The Digital Elite: Smartphones are getting huge. Can the average-size thumb keep up?” (discusses my study on thumb size and mobile touch screens.
22. Aug 13, 2012 **Fact Witness in Apple v. Samsung trial covered broadly**
<http://blogs.wsj.com/digits/2012/08/13/samsungs-first-witnesses-push-prior-art-after-minor-victory/>
http://news.cnet.com/8301-13579_3-57492486-37/samsung-goes-on-offense-aims-to-bust-apple-patents/
<http://arstechnica.com/tech-policy/2012/08/apple-rests-patent-case-as-judge-tosses-three-samsung-phones-from-trial/>
<http://www.forbes.com/sites/connieguglielmo/2012/08/13/apple-says-it-complained-about-samsung-copying-in-2010-live-blog/>
<http://allthingsd.com/20120813/judge-refuses-to-toss-most-of-apples-suit-against-samsung/>
<http://www.theverge.com/2012/8/13/3240686/samsung-apple-patents-launchtile-diamonddtouch-table>
<http://www.eweek.com/c/a/Mobile-and-Wireless/Samsung-Begins-Presenting-Case-in-Apple-Patent-Infringement-Trial-162894/>
23. May 7, 2012 **WAMU Kojo Nnamdi Show**
“Free Online Courses - Higher Ed of the Future?” (guest on hour show)
<http://thekojoannamdishow.org/shows/2012-05-07/free-online-courses-higher-ed-future>
24. January 24, 2012 **Washington Post** by Bonnie Berkowitz and Alberto Cuadra
“Touchscreens: How they work”, I am the primary source
http://www.washingtonpost.com/national/health-science/touchscreens-how-they-work/2012/01/23/gIQAZHIA MQ_graphic.html
25. June 28, 2011 **New Scientist** by Jim Giles
“The man-machine: Harnessing humans in a hive mind”, discussed MonoTrans.
<http://www.newscientist.com/article/mg21028181.900-the-manmachine-harnessing-humans-in-a-hive-mind.html>
26. June 14, 2011 **WAMU Kojo Nnamdi Show**
“A Shift in Cloud Computing” (guest on hour show)
<http://thekojoannamdishow.org/shows/2011-06-14/shift-cloud-computing>
27. May 13, 2011 **Discovery News** by Alyssa Danigelis
“Tech Turns Air into a Multi-touch screen”, quoted.
<http://news.discovery.com/tech/touch-screen-technology-zero-touch-110513.html>
28. April 28, 2011 **Converge Magazine** by Tanya Roscorla
“Music Apps Take Top Prizes in U of Maryland Contest”, discusses mobile app from the University of Maryland mobile app contest that I ran.

29. April 26, 2011 **Voice of America** by Do Noghte Sefr (in Persian)
 “Mobility Initiative ابتکار تحرک”
<http://www.youtube.com/donoghtesefr>
30. April 24, 2011 **Washington Post**
 “Business Rx: A jukebox democracy in search of voters”, discusses mobile app from the University of Maryland mobile app contest that I ran.
31. February 25, 2011 **New Scientist** by Jim Giles and Jacob Aron
 “Crowdsourced translations get the word out from Libya”, discusses my MonoTrans project
<http://www.newscientist.com/article/dn20174-crowdsourced-translations-get-the-word-out-from-libya.html>
32. February 9, 2011 **DiamondBack, UMD Student Newspaper**
 “Creativity at her fingertips” by Erin Egan
http://www.diamondbackonline.com/news/article_f7143841-131c-5695-a310-42596336345d.html
33. February 8, 2011 **WAMU Kojo Nnamdi Show**
 “A Revolution in Tech Jobs? The Murky World of Online Labor” (hour show with me as only guest)
<http://thekojoannamdishow.org/shows/2011-02-08/revolution-tech-jobs-murky-world-online-labor>
34. November 23, 2010 **WAMU Kojo Nnamdi Show**
 “The End of Email and other Tech Prognostications”
 (hour show with me and one other guest)
<http://thekojoannamdishow.org/shows/2010-11-23/end-email-and-other-tech-prognostications>
35. November 9, 2010 **MIT Technology Review** by Erica Naone
 “Gesturing at Your TV Isn’t Ready for Prime Time” quotes me discussing Microsoft Kinect
<http://www.technologyreview.com/computing/26691/>
36. October 27, 2010 **MIT News** by Larry Hardesty
 “Programming crowds” quotes me in analyzing Rob Miller’s work
<http://web.mit.edu/newsoffice/2010/programming-crowds-1027.html>
37. October 2, 2010 **News 8 TV Austin**
 “App Wrap: eReading for Kids” discussing StoryKit app for iPhone
<http://www.news8austin.com/content/headlines/274426/app-wrap--ereading-for-kids>
38. August 24, 2010 **WAMU Kojo Nnamdi Show**
 “Automating Language Translation: The Future of a Unified Internet?” discussing my research and the general topic of crowdsourcing and translation.
<http://thekojoannamdishow.org/shows/2010-08-24/automating-language-translation-future-unified-internet>
39. May 4, 2010 **Technology Review**
 “Redesigning the Web for Touch Screens”
http://www.technologyreview.com/printer_friendly_article.aspx?id=25236&channel=computing§ion=
40. April 2010 **FutureGov Magazine** by Kelly Ng, Singapore
 Description of ICDL for iPad application
41. April 17, 2010 **ABC 7 News, WLS-TV Chicago**
 “Chicago Students Face Off in Battle of Books”
<http://abclocal.go.com/wls/story?section=news/local&id=7392055>
42. April 8, 2010 **Publisher’s Weekly**
 “The iPad Meets the Children’s Book”
http://www.publishersweekly.com/article/455914-The_iPad_Meets_the_Children_s_Book.php?nid=2788&source=link&rid=16811925
43. April 3, 2010 **The Independent**, London
 “Children’s book iPad apps offer multilingual tales, long-distance bedtime stories”
<http://www.independent.co.uk/arts-entertainment/books/childrens-book-ipad-apps-offer-multilingual-tales-longdistance-bedtime-stories-1934847.html>

44. January 28, 2010 **Baltimore Sun**, Baltimore, MD
“Apple unveils tablet computer” by Gus G. Sentementes
<http://www.baltimoresun.com/business/technology/bal-bz.ipad28jan28,0,5120135.story>
45. December 2009 **FutureGov Magazine** by Kelly Ng, Singapore
“Inside the Teacher’s Lounge” covered Ben’s keynote UNESCO talk in Hangzhou, China
46. November 20, 2009 **World Bank Blog**
<http://blogs.worldbank.org/edutech/icdl-1>
47. November 2, 2009 **Technology Review.com**
“First Test for Election Cryptography”
<http://www.technologyreview.com/web/23836/>
48. December 23, 2008 **WAMU Kojo Nnamdi Show**, Washington, DC
“Computer Guy”
49. September 18, 2008 **America.gov**
“Kids Help Design International Children’s Digital Library” by Jeffrey Thomas
<http://www.america.gov/st/educ-english/2008/September/200809181459141CJsamohT0.2946283.html>
Also published in **gulf-times.com** in Qatar
http://www.gulf-times.com/site/topics/article.asp?cu_no=2&item_no=245520&version=1&template_id=46&parent_id=26
50. April 25, 2008 **C|Net News.com**
“The Six Secrets of Mobile Computing Success” by Tom Krazit
51. February 8, 2008 **WYPR Maryland Morning**, Baltimore, MD
10 minute radio interview about usability of voting systems
52. January 30, 2008 **KCSN News** by Jessica Caldera
Short radio interview about usability of voting systems
53. January 29, 2008 **Technology Review.com** by Erica Naone
“Voting with (Little) Confidence”
<http://www.technologyreview.com/Infotech/20122/>
54. January 28, 2008 **ScienceDaily.com**
“Touch Screen Voting A Hit; Critics Miss Mark on Security, Study Says”
<http://www.sciencedaily.com/releases/2008/01/080123170954.htm>
55. January 26, 2008 **Newsweek** by Barrett Sheridan
“Like a Super Hero: Humans weren’t made for scrolling and searching. We were made for zooming.”
<http://www.newsweek.com/id/105532/output/print>
56. January 25, 2008 **Government Executive.com** by Gautham Nagesh
“Voters confused by e-voting machines, study finds”
<http://www.govexec.com/dailyfed/0108/012508n1.htm>
57. January 22, 2008 **WAMU Kojo Nnamdi Show**, Washington, DC
Thirty minute segment on “Technology User Frustrations – Spotlight on Cellphones”
<http://wamu.org/programs/kn/08/01/22.php#19083>
58. January 16, 2008 **KUOW, The Conversation**, Seattle, WA
The guest on an hour-long call-in talk show on the future of cellphones.
<http://www.kuow.org/defaultProgram.asp?ID=14159>
59. October 30, 2007 **WAMU Kojo Nnamdi Show**, Washington, DC
“Computer Guy”, talking about software reliability
60. August 7, 2007 **WAMU Kojo Nnamdi Show**, Washington, DC
“Computer Guy”, current tech events, iPhone, Tivo, and listeners email and calls
<http://wamu.org/programs/kn/07/08/07.php>

61. June 13, 2007 **New York Times** by John Markoff
"That iPhone has a keyboard, but it's not mechanical"
<http://www.nytimes.com/2007/06/13/technology/13phone.ready.html>
62. June 5, 2007 **WAMU Kojo Nnamdi Show**, Washington, DC
"Computer Guy", Microsoft Surface, cell phone problems, and listeners email and calls
<http://wamu.org/programs/kn/computerguys/07/cg070605.php>
63. April 3, 2007 **WAMU Kojo Nnamdi Show**, Washington, DC
"Computer Guy", Physical and virtual convergence, and listeners email and calls
<http://wamu.org/programs/kn/computerguys/07/cg070403.php>
64. March 17, 2007 **Science News** weekly magazine, by Ivars Peterson
"Games Theory", several quotes by me on article about von Ahn's games.
<http://www.sciencenews.org/articles/20070317/bob9.asp>
65. January 18, 2007 **Wall Street Journal**, by Jeremy Wagstaff
"The March of Time", covering DateLens
66. December 31, 2006 **Associated Press**, by Brian Bergstein
"Low-Cost Laptop Could Transform Learning"
<http://www.washingtonpost.com/wp-dyn/content/article/2006/12/31/AR2006123100380.html>
AP story picked up in 175 locations, including:
- Washington Post
 - BusinessWeek
 - Chicago Tribune
 - Boston Globe
 - ABC News
 - CNN
 - FOX News
 - Eyewitness News
 - Forbes
 - BBC News, UK
 - Sydney Morning Herald, Australia
 - The Age, Australia
 - Economic Times, India
67. December 5, 2006 **WAMU Kojo Nnamdi Show**, Washington, DC
"Computer Guy", Vista, holiday tech gifts, and listeners email and calls
<http://www.wamu.org/programs/kn/06/12/05.php#12452>
68. September 22, 2006 **Salon.com** by Katharine Mieszkowski
"Written out of the story", discussing write-in votes using the Hart eSlate system
http://www.salon.com/news/feature/2006/09/22/delay_write_in/index_np.html
69. May 2, 2006 **WAMU Kojo Nnamdi Show**, Washington, DC
"Computer Guy – life in the fast lane", discussing broadband at home
<http://www.wamu.org/programs/kn/06/05/02.php#10686>
70. April 23, 2006 **Pittsburgh Post-Gazette** by Mark Roth
"Experts see computers getting bigger and smaller at the same time"
<http://www.post-gazette.com/pg/06113/684425-96.stm>
71. March 9, 2006 **Baltimore Sun** by Mike Himowitz
"PhotoMesa a nifty way to organize photographs"
<http://www.baltimoresun.com/technology/bal-bz.himowitz09mar09,0,3518441.column>
72. February 21, 2006 **WAMU Kojo Nnamdi Show**, Washington, DC
"Wiki Sites"
<http://www.wamu.org/programs/kn/06/02/21.php#10383>

73. February 3, 2006 **Baltimore Sun** by Frank D. Roylance
 "Spreading the e-word" (quoted about Sony's new ebook reader e-paper)
<http://www.baltimoresun.com/news/health/bal-hs.eink03feb03,1,3010066.story>
74. January 13, 2006 **Washington Business Journal** by Jennifer Nycz-Conner
 "What was I doing, again?"
<http://www.bizjournals.com/washington/stories/2006/01/16/focus1.html>
75. November 18, 2005 **c|net News.com** by Stefanie Olsen
 "The 'millennials' usher in a new era" (about ICDL)
http://news.com.com/The+millennials+usher+in+a+new+era/2009-1025_3-5944666.html?tag=nefd.lede
76. September, 2005 **Chicago Parent** by Jane Huth
 "Screen time=reading time"
<http://www.chicagoparent.com/main.asp?SectionID=10&SubSectionID=32&ArticleID=529&TM=67415.63>
77. September 22, 2005 **Delphos Herald**
 "Site Seeing"
<http://www.delphosherald.com/page2.php?story=8655&archive=>
78. April, 2005 **Chicago Parent** by Judy Belanger
 "Explore the world by turning pages"
<http://www.chicagoparent.com/main.asp?SectionID=10&SubSectionID=33&ArticleID=306&TM=67012.6>
79. April 5, 2005 **EE Times** by Nicolas Mokhoff
 "Microsoft touts 'thumb-as-stylus' interface progress"
<http://www.eetimes.com/showArticle.jhtml;jsessionid=NHVZOY3L1AQWUQSNDBGCKH0CJUMKJVN?articleID=160500234>
 Also picked up by **Information Week**
80. March 10, 2005 **folha equilibrio (Brazil)** by Sérgio Rizzo
 "Cibertentações" (article about computer-based distraction, describing some of my work.)
81. March, 2005 **Radio City (Ecuador) / BBC Affiliate**
 5 minute radio interview (performed in English, translated into Spanish)
82. March 1, 2005 **The Statesman (India)** by Stanley Theodore
 "Webcrawler: All About E-Attention Deficit Disorder"
<http://www.thestatesman.net/page.news.php?clid=24&theme=&ursess=1&id=70119>
83. February 28, 2005 **Washington Times** by Ann Geracimos
 "New Chapter in Online Books"
<http://washingtontimes.com/metro/20050227-101610-1015r.htm>
84. February 18, 2005 **North Carolina State Technician** by Danny Frye
 "Research targets computer distraction such as AIM"
<http://technicianonline.com/story.php?id=011167>
85. February 13, 2005 **Bloomberg Radio, "Simply Put"** by Michael Goldman & Tom Moroney
 9 minute interview discussing attention and interruptions on the computer
86. February 10, 2005 **New York Times** by Katie Hafner
 "You There, at the Computer: Pay Attention"
<http://www.nytimes.com/2005/02/10/technology/circuits/10info.html>
 Also picked up by
 - a. **International Herald Tribune (France)**
 - b. **ABS CBN News (Philippines)**
 - c. **CNET News.com**
 - d. **Spartanburg Herald (SC)**
 - e. **Wilmington Morning Star (NC)**
 - f. **Minneapolis Star Tribune (MN)**
87. Nov 11, 2004 **Maryland Public Television** by Frank Batavick
 "Message in a Bottle", Bob the vid-tech guy 30 minute special including 5 minute segment of Allison Druin and me explaining email and instant messaging to children.

88. May 7, 2004 **Baltimore Sun** by Elizabeth Schuman
"Research Lab Crosses Cultures"
89. October 20, 2003 **The Feature** by Jeff Goldman
"A View Into Your Phone"
<http://www.thefeature.com/article?articleid=100146>
90. October 10, 2003 **The Independent** by Charles Arthur
"Treemap Offers a New Way of Visualizing Complex Data. Nothing Beats it for Simplicity and the "wow!" effect when you first use it."
<http://news.independent.co.uk/digital/features/story.jsp?story=451099>
91. July, 2003 **Higher Learning – Technology Serving Education**
"Web Sites: The International Children's Digital Library"
92. July 7, 2003 **Pittsburgh Post-Gazette** by Karen MacPherson
"Web Site Invites Children to Read Online: International Children's Digital Library Caters to Young Around the Globe"
93. June 15, 2003 **San Antonio News 9** by Michael Garfield
"Website a Fun Way to Encourage Kids to Read"
94. April 11, 2003 **San Jose Mercury News** by Dan Gillmor
"Designing New Handhelds to Improve Interactions Between Humans, Computers"
Also picked up by **New Jersey Star Ledger (NJ)**
95. February 12, 2003 **Tech TV Live** by Peter Barnes
Also on **abcnews.com**
"Easy Online Reader: An International Online Nook for Children's Books"
96. January 17, 2003 **Voice of America TV** by Larry Clamage
4.5 minute video of children's design team focusing on ICDL
97. December, 2002 **Sun's Java Website**
ICDL Linked to from <http://java.sun.com/industry/>
98. December 17, 2002 **Associated Press**
Picture of Ben Bederson w/ Bill Gates at Microsoft about DateLens project
 - a. **Seattle Post Intelligencer (WA)**
 - b. **Times Daily (AL)**
 - c. **Sarasota Herald-Tribute (FL)**
 - d. **Naples Daily (FL)**
99. December 16, 2002 **Baltimore Sun** by Alex MacGillis
"Two Study Intricacies of Voting Machines"
Also picked up by AP, excerpts in:
 - a. **USA Today**
 - b. **WMAR (MD)**
 - c. **WBAL-TV (MD)**
 - d. **News Channel 8.net**
100. December 12, 2002 **Philadelphia Inquirer** by Joyce Kasman Valenza
"tech.life @ school | New virtual library offers children much"
101. December 10, 2002 **Le Monde (France)** by Chantal Dussuel
"Livres en ligne pour internautes on herbe"
102. December 10, 2002 **Christian Science Monitor**
News in Brief – ICDL
103. December 5, 2002 **New York Times** by Lisa Guernsey
"Online Library Project Plans a Digital and Cultural Trove for Children"
104. December 1, 2002 **Die Burger (South Africa)**
"Paragrawe"

105. November 30, 2002 **The Vancouver Sun (Canada)**
"Book Endz: Free Kid's Books"
106. November 30, 2002 **La Vanguardia (Spain)**
"Nuevos Proyectos: TalentMatch.com, International Children's Digital Library SoapCity"
107. November 27, 2002 **USA Today** by Karen Thomas
"World Books, Children Click"
108. November 27, 2002 **The Ohama World-Herald**
"TechBytes: Digital Library Links Kids to Other Cultures"
109. November 27, 2002 **Education Week**
"Internet Library for Children Invites the World to Read"
110. November 26, 2002 **The Gold Coast Bulletin (Australia)**
"The Net: News Bytes"
111. November 25, 2002 **intern.de (Germany)**
"Kinderbibliothek"
112. November 25, 2002 **Seattle Post-Intelligencer**
"Kids Books Now Online"
113. November 21, 2002 **AP Wire** by Sam Hannel
"Web Site to Give Free Access to Children's Books"
a. **ABCNews.com**
b. **The Age (Australia)**
c. **Ananova.com (UK)**
d. **Asia Pacific Media Network**
e. **The Baltimore Sun**
f. **Globetechnology.com (Canada)**
g. **KTVU 2 News (Bay Area)**
h. **Indiana Gazette**
i. **MSNBC.com**
j. **Newsday.com**
k. **Rapid City Journal**
l. **San Jose Mercury News**
m. **Sarasota Herald-Tribune (Florida)**
n. **Santa Fe New Mexican**
o. **Seattle Post-Intelligencer**
p. **Sydney Morning Herald (Australia)**
q. **Waterloo/Cedar Falls**
r. **WFTV (Florida)**
114. November 21, 2002 **24ur.com (Slovenia)**
"Knjige brezplačno na internetu"
115. November 21, 2002 **Courier Mail (Australia)**
116. November 21, 2002 **United Press International (UPI)**
"New Online Library for Kids Launched"
117. November 21, 2002 **Washington Post (.COM Column)** by Leslie Walker
"A Library for Young Browsers"
a. **International Herald Tribune**
b. **SFGate.com (San Francisco)**
118. November 21, 2002 **Internet Magazine**
"Site Offers Free Access to Kids Books"
119. November 20, 2002 **Magasinet (Norway)**
"Lager verdensbarnebibliotek"
120. November 20, 2002 **Fox 5 TV News** by John Henrehan
Live 1.5 minutes on 5:00pm news

121. November 18, 2002 **All Things Considered on NPR** by David Kestenbaum
"Library for Kids Goes Online" (11 minutes)
122. October, 2002 **Digital Computer Magazine**
Review of PhotoMesa
123. September 23, 2002 **ComputerSweden** by Anders Lotsson
"Square Zoom in the Hand"
124. August 19, 2002 **Ohio Repository** by Joan Renner
"Ensuring validity is a big concern in electronic voting"
125. August 18, 2002 **Ohio Repository** by Joan Renner
"Voting test shows pros, cons"
126. August 12, 2002 **eWeek**, by Anne Chen
"DateLens Stays Organized in C#"
127. August 5, 2002 **Grid Today**
"Bill Gates Highlights Academic Collaboration Key to Future"
128. July 29, 2002 **eWeek**, by Anne Chen
"Gates: .NET Won't Happen Overnight"
129. July 29, 2002 **InfoWorld**, by Matt Berger
"Gates, academics joins on security, shared source"
130. July 29, 2002 **NetworkWorldFusion**, by Matt Berger
"Gates, academics joins on security, shared source"
131. May 9, 2002 **Washington Post (.COM Column)**, by Leslie Walker
"A Visual Rather Than Verbal Future"
132. May 9, 2002 **Washington Post Website**, by Eleanor Hong
"Human-Computer Interaction Lab at the University of Maryland" – Interview with me, Allison Druin, and Ben Shneiderman.
133. August 1, 2001 **Sun Microsystems Java Website**, by Jon Byous
"View With a Zoom: Browse and Zoom Digital Images in One Application" describes PhotoMesa and my treemap algorithms.
134. July 2, 2001 **ComputerWorld** by Sami Lais
"Treemaps Bloom" describes my photo browser application PhotoMesa which uses Jazz
135. June 4, 2001 **ComputerWorld** by Sami Lais
"Software for the 4th Dimension" describes TimeSearcher which uses Jazz
136. August 12, 2000 **New Scientist Magazine**
"Maximizing Minimalism", an article Zoomable User Interfaces, KidPad, and Fisheye Menus
137. Fall, 2000 **College Park**, by Tom Ventsias
"Zooming Ahead", an article about Zoomable User Interfaces
138. June 26, 2000 **ComputerWorld**, by Sami Lais
"Zoomin' Ahead", an article about Zoomable User Interfaces and Fisheye Menus
139. September, 1999 **Sun Microsystems Java Website**, by Jon Byous
An interview with me under their face-to-face column.
140. December 21, 1998 **Baltimore Sun**
Reprinted in New York Newsday, Dec. 30, 1998
"Children first: In designing toys, one lab turns to the experts in the field."
141. April 4, 1997 **The Chronicle of Higher Education** by Jeffrey R. Young
"New Ways of Organizing Data Could Change Nature of Computers"
142. February, 1997 **Nov'Art**
a French Magazine sponsored by Art3000, a society devoted to integration of technology and art.

143. October 23, 1996 **The New York Times Online (CyberTimes)** by Ashley Dunn
 "Seeing the Forest of Knowledge for the Trees of Data"

144. June, 1995 **The X Advisor**, by Gary Welz
 "Peripheral Visions: Zooming Through Information Space on Pad++"

3. TEACHING AND ADVISING

A. COURSES TAUGHT

i. General

Fall 2015	<i>Paths to CS</i> (CMSC198C/D/E) – 96 Students
Fall 2013	<i>Paths to CS</i> (CMSC198C/D/E, 1-3 credits) – 36 Students CMSC198C Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.6 (College comparison: 2.8) Effective teacher: 3.9 (College comparison: 2.9) CMSC198D Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.8 (College comparison: 2.8) Effective teacher: 4.0 (College comparison: 2.9)
Fall 2012	<i>Tech for CS Startups – Team taught</i> (CMSC 498E, 1 credit) – 19 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.0 (College comparison: 3.2) Effective teacher: 3.7 (College comparison: 2.9)
Spring 2012	<i>Reading Seminar: Future of HCI – Team taught</i> (CMSC 838B, 1 credit) – 13 Students
Fall 2011	<i>Social Computing Seminar</i> (CMSC 838B, 3 credits) – 20 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.7 (College comparison: 3.2) Effective teacher: 3.8 (College comparison: 3.2)
Fall 2011	<i>Intro to Human-Computer Interaction</i> (CMSC 434) – 37 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.5 (College comparison: 3.2) Effective teacher: 3.5 (College comparison: 3.0)
Spring 2011	<i>Intro to Human-Computer Interaction</i> (CMSC 434) – 42 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 2.9 (College comparison: 3.2) Effective teacher: 3.0 (College comparison: 3.0)
Fall 2010	<i>Reading Seminar in Human Computation</i> (CMSC 838B, 1 credit) – 9 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.8 (College comparison: 3.2) Effective teacher: 3.9 (College comparison: 3.5)
Fall 2010	<i>Digital Cultures and Creativity</i> (HDCC 105, co-taught) – 56 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 2.4 (College comparison: 2.5) Effective teacher: 3.2 (College comparison: 3.4)
Spring 2010	<i>Object-Oriented Programming I</i> (CMSC 131H) – 10 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.8 (College comparison: 2.9) Effective teacher: 4.0 (College comparison: 3.0)
Fall 2009	<i>Intro to Human-Computer Interaction</i> (CMSC 434) – 39 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.3 (College comparison: 3.2) Effective teacher: 3.5 (College comparison: 3.0)

Spring 2008	<i>Intro to Human-Computer Interaction</i> (CMSC 434) – 32 Students Student Evaluations (Scale: 0 Poor – 4 Excellent) Students “learned a lot”: 3.1 (College comparison: 3.1) Effective teacher: 3.2 (College comparison: 3.0)
Spring 2005	<i>Developing User Interfaces</i> (CMSC 498B) – 14 Students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.6 Instructor's teaching: 3.8
Spring 2004	<i>Object Oriented Programming I</i> (CMSC 131) – 75 Students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.0 Instructor's teaching: 2.9
Spring 2003	<i>Information Visualization</i> (CMSC 828B) – 17 Students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.1 Instructor's teaching: 3.4
Spring 2002	<i>Developing User Interfaces</i> (CMSC 498B) – 25 Students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.6 Instructor's teaching: 3.9
Fall 2001	<i>Human Computer Interaction</i> (CMSC 434/828S) – 50 Students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.5 Instructor's teaching: 3.6
Spring 2001	<i>Information Visualization</i> Taught by Catherine Plaisant and Ben Shneiderman due to my illness
Fall 2000	<i>Human Computer Interaction</i> (CMSC 434/828S) – 50 Students No evaluation due to my illness
Spring 2000	<i>Computer Graphics</i> (CMSC 427) – 45 students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.4 Instructor's teaching: 3.3
Fall 1999	<i>Human Computer Interaction</i> (CMSC 434/828S) – 44 Students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.9 Instructor's teaching: 3.9
Spring 1999	<i>Zoomable User Interfaces</i> (CMSC 838B) – 15 Students
Fall 1998	<i>Software Engineering</i> (CMSC 435) – 45 students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.6 Instructor's teaching: 3.5
Spring 1998	<i>Computer Graphics</i> (CMSC 427) – 24 students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.7 Instructor's teaching: 3.7
Spring 1997	<i>Software Engineering</i> (CS460) @ UNM (Didn't receive evaluations)
Fall 1996	<i>Introduction to Data Structures</i> (CS251) - 100 Students @ UNM (Didn't receive evaluations)

Spring 1996	<i>Human Computer Interaction (CS462) @ UNM</i> Student Evaluations (ICES scale: 1 poor - 6 excellent) Course content: 5.2 Instructor: 5.5 Course in general: 5.2
Spring 1996	<i>Computers & Society (CS491/591) @ UNM</i> Introduced this course for the first time in the CS dept. 1 credit seminar Student Evaluations (ICES scale: 1 poor - 6 excellent) Course content: 4.9 Instructor: 4.8 Course in general: 4.8
Spring 1996	Taught one week <i>Computer Ethics @ UNM</i> section of a new five week introductory class on computing at UNM (CS101). Did this two times.

ii. Specialized

Fall 2016	<i>Learning Analytics Research Group (TLTC 708)</i> – 1 credit, 3 students, 20 participants
Spring 2016	<i>Learning Analytics Research Group</i> – 0 credits, 30 participants
Fall 2015	<i>Learning Analytics Reading Group</i> – 0 credits, 20 participants
Spring 2015	<i>Learning Analytics Reading Group</i> – 0 credits, 20 participants
Fall 2003	<i>Readings in HCI Seminar (CMSC 838B)</i> – 1 credit, 13 students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.9 Instructor's teaching: 3.8
Fall 2001	<i>Readings in HCI Seminar (CMSC 838B)</i> – 1 credit, 12 students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.8 Instructor's teaching: 3.8
Fall 2000	<i>Readings in HCI Seminar (CMSC 838B)</i> – 1 credit, 15 students
Fall 1999	<i>Readings in HCI Seminar (CMSC 838B)</i> – 1 credit, 6 students
Spring 1999	<i>Zoomable User Interfaces (CMSC 838B)</i> – 15 students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 3.5 Instructor's teaching: 3.2
Fall 1998	<i>Readings in HCI Seminar (CMSC 838B)</i> - 1 credit, 9 students Student Evaluations (Scale: 0 Poor - 4 Excellent) Course quality: 4.0 Instructor's teaching: 3.8

F. ADVISING RESEARCH DIRECTION

i. Undergraduate

Independent Study Advisor:

2013	Andrej Resevic – new Paths to Computer Science course
2012 – 2013	Danny Michaelis – new Paths to Computer Science course
2010 – 2011	Jonathan Speiser Worked on tournament programmer in educational technologies group
2003-2004	Christian Klein Studied effect of animation when scrolling while reading

2000 – 2002	Rob Sherman Customized undergraduate degree in HCI & Law – focused on voting systems
Fall 2001	Steve Betten Visualizations for real-time monitoring
Spring 2000/ Fall 2000	Beth Weinstein Geographical interfaces for digital library for children
Spring 1999	Nuno Pereria End-user customizable systems
Spring 1996	Philip Eckenroth and Mike Tipping Built a 3D modeling system
Fall 1995	Josh Saiz Novel searching techniques in Pad++ (dynamic indicators)

ii. Master's

Thesis Committee:

2005	Julie Staub in Applied Math. Research on Chaum's scheme for encrypted voter receipts.
1999	George Ziets in Psychology. Advisor is Kent Norman. Research is on gauge design for multiple orders of magnitude.

Independent Study Advisor:

2013	Alina Goldman (iSchool) – Motivation in a self-paced computer science course
2012	Adam O'Sullivan – Computer vision for relative phone location
2000	Stephanie Thibeault (Dance Dept.) Technology and dance
1996	Ellen Killion Built a software library access tool in Pad++
1995	Tom Claus Worked on KidPad
1995	Sam Cancilla Shared calendar application in Pad++
1995	David Rogers Tossing for user interfaces
1995	Richard Bleakman Interfaces for teaching genetic algorithms
1995	Tom Claus Worked on Pad++

iii. Doctorate

Dissertation Advisor:

2017	Tak Yeon Lee, Ph.D. (2017) in Computer Science. "Toward Symbiotic Human-AI Interaction Focusing on Programming by Example". Currently at Adobe Research.
2016	Adil Yalcin, Ph.D. (2016) in Computer Science. "Towards Rapid, Effective, and Expressive Visual Data Exploration". Currently founding startup to commercialize his dissertation research - https://keshif.me/.

- 2014 Alex Quinn, Ph.D. (2014) in Computer Science. “Crowdsourcing Decision Support: Frugal Human Computation for Efficient Decision Input Acquisition”. **Currently Assistant Professor of ECE at Purdue.**
- 2012 Chang Hu, Ph.D. (2012) in Computer Science. “Crowdsourced Monolingual Translation”. **Currently at Microsoft.**
- 2007 Amy Karlson, Ph.D. (2007) in Computer Science. Research on personal information management. **Winner of Microsoft Graduate Fellowship. Two CHI papers nominated for best paper. Worked at Microsoft Research for several years.**
- 2006 Bongshin Lee, Ph.D. (2006) in Computer Science. “Interactive Visualization for Trees and Graphs”. **Co-first place winner of IEEE InfoVis 2004 Contest. Currently at Microsoft Research.**
- 2005 Hilary Hutchinson, Ph.D. (2005) in Computer Science. “Children’s Interface Design for Searching and Browsing”. **Awarded UMD CS Minker Fellowship. Awarded NSF Graduate Fellowship. Currently at Google.**
- 2005 Bongwon Suh, Ph.D. (2005) in Computer Science. “Image Management Using Pattern Recognition”. **Worked at PARC, then at Adobe Research, now Assoc. Prof. at Seoul National University (Korea).**
- 2003 Lance Good, Ph. D. (2003) in Computer Science. “Zoomable User Interfaces for the Authoring and Delivery of Slide Presentations”. **Worked at PARC as researcher for several years, now at Google.**
- 2003 Juan Pablo Hourcade, Ph.D. (2003) in Computer Science. “User Interface Technologies and Guidelines to Support Children’s Creativity, Collaboration, and Learning”. **Currently Assoc. Prof. of Computer Science at the University of Iowa.**
- 1998 Jason Stewart, Ph.D. (1998) in Computer Science at UNM. Research is on collaborative user interfaces for children using Pad++. **Awarded 1995 DEC Fellowship at UNM.**
- Dissertation Committee:**
- 2020 Virginia Leigh Byrne (2020) in Education. Advisor is Diane Jass Ketelhut. “The Chilling Effect of Cyberbullying on Women’s Social Presence in Online Discussions”
- 2019 Zhenpeng Zhao (2019) in Computer Science. Advisor is Niklas Elmqvist. “Data-Driven Storytelling for Casual Users”
- 2016 Uran Oh, Ph.D. (2016) in Computer Science. Advisor is Leah Findlater. “Accessible On-Body Interaction for People with Visual Impairments”
- 2016 Kotaro Hara, Ph.D. (2016) in Computer Science. Advisor is Jon Froehlich. “Scalable methods to collect and visualize sidewalk accessibility data for people with mobility impairments”
- 2014 Aaron Dinin, Ph.D. (2014) in English. Advisor is Martha Nell Smith. “Hacking Literature: A Study in the Technology of Language.”
- 2014 Brian Danielak, Ph.D. (2014) in College of Education. Advisor is Andy Elby. “How Electrical Engineering Students Design Computer Programs.”
- 2013 Irene Eleta Mogollon, Ph.D. (2013) in iSchool. Advisor is Jen Golbeck. “Multilingual use of Twitter: Language Choice and Language Bridges in a Social Network.”
- 2013 John Alexis Guerra Gomez, Ph.D. (2013) in Computer Science. Advisor is Ben Shneiderman. “Exploring Differences in Multivariate Datasets Using Hierarchies: An Interactive Information Visualization Approach”
- 2013 Sureyya Tarkan, Ph.D. (2013) in Computer Science. Advisor is Ben Shneiderman. “Interactive Visual Displays for Results Management in Complex Medical Workflows”
- 2010 Taowei David Wang, Ph.D. (2010) in Computer Science. Advisor is Ben Shneiderman. “Interactive Visualization Techniques for Searching Temporal Categorical Data”

2009 Jerry Fails, Ph.D. (2009) in Computer Science. Advisor is Allison Druin. "Mobile Collaboration for Young Children"

2008 Chunyuan Liao, Ph.D. (2008) in Computer Science. Advisor is Francois Guimbretiere. "PapierCraft: A New Paper-Based Interface To Support Interaction With Digital Documents"

2008 Aleks Aris, Ph.D. (2008) in Computer Science. Advisor is Ben Shneiderman. "Visualization & Exploring Networks Using Semantic Substrates"

2008 Adam Perer, Ph.D. (2008) in Computer Science. Advisor is Ben Shneiderman. "Integrating Statistics and Visualization to Improve Exploratory Social Network Analysis"

2008 Walky Rivadeneira, Ph.D. (2008) in Psychology. Advisor is Kent Norman. "Tag Clouds: How Format and Categorical Structure Affect Categorization Judgement"

2007 Matthias Mayer, Ph.D. (2007) in Dept. Informatik, Universität Hamburg where I was an outside reader. Advisor is Prof. Dr. Arno Rolf. "Visualizing Web Sessions".

2007 Leslie Eugene Chipman, Ph.D. (2007) in Computer Science. Advisor is Allison Druin. "Collaborative Technology for Young Children's Outdoor Education".

2006 Haibin Ling, Ph.D. (2006) in Computer Science. Advisor is David Jacobs. "Techniques for Image Retrieval: Deformation Insensitivity and Automatic Thumbnail Cropping".

2005 Jinwook Seo, Ph.D. (2005) in Computer Science. Advisor is Ben Shneiderman. "Information Visualization Design for Multidimensional Data: Integrating the Rank-By-Feature Framework with Hierarchical Clustering".

2005 Jen Golbeck, Ph.D. (2005) in Computer Science. Advisor is Jim Hendler. "Computing and Applying Trust in Web-Based Social Networks".

2003 Harry Hochheiser, Ph.D. (2003) in Computer Science. Advisor is Ben Shneiderman. Research on interfaces for time-series data.

2002 Jenny Sterling, Ph.D. (2002) in Music. Advisor is Laszlo Payerle. Research on educational software she developed to teach undergrad music form. She won the campus award on educational technologies from the Center for Teaching Excellence.

2001 Mike Beynon, Ph.D. (2001) in Computer Science. Advisor is Joel Saltz. Research on filter-based programming model for grid computing.

2001 Angela Boltman, Ph.D. (2001) in Education, Advisor is Allison Druin. Research on KidPad's effect on children's storytelling.

2001 Jeff Heflin, Ph.D. (2001) in Computer Science. Advisor is Jim Hendler. Research on the semantic web.

2001 Egemen Tanin, Ph.D. (2001) in Computer Science. Advisor is Ben Shneiderman. Research on browsing large online databases using exploratory overviews.

2000 Vasanth Philomin, Ph.D. (2000) in Computer Science. Advisor is Larry Davis. Research on real-time computer vision for in-car navigation and warning systems.

2000 Byoung-Kee Yi, Ph.D. (2000) in Computer Science. Advisor is Christos Faloutsos. Research on similarity search and data mining in time-sequence databases.

2000 Chris North, Ph.D. (2000) in Computer Science. Advisor is Ben Shneiderman. Research on tight coupling between different visualizations.

1999 Richard Potter, Ph.D. (1999) in Computer Science. Advisor is Ben Shneiderman. Research on pixel data access for end-user customization.

1999 Harald Winroth, Ph.D. (1999) in Computer Science at the Royal Institute of Technology (KTH) in Stockholm, SWEDEN. I was invited to be the *opponent*. Advisor is Jan-Olof Eklundh Research on dynamic projective geometry.

- 1999 Ismail Haritaoglu, Ph.D. (1999) in Computer Science. Advisor is Larry Davis. Research on real-time computer vision for tracking people.
- 1999 Zhijun Zhang, Ph.D. (1999) in Computer Science. Advisors are Vic Basili and Ben Shneiderman. Research on perspective-based usability inspection techniques.
- 1998 Elaine Raybourne, Ph.D. (1998) in Communication and Journalism at the University of New Mexico. Advisor is Ed Rogers. Dissertation is on the use of MUDs for anonymous role-playing to teach about issues of culture, identity, and power.
- 1998 Fan-Tao Pu, Ph.D. (1998) in Computer Science, Advisor is David Mount. Research on data structures for view-independent 3D object visibility detection.
- 1998 Eser Kandogan, Ph.D. (1998) in Computer Science, Advisor is Ben Shneiderman. Research on a technique for window management – elastic windows.
- 1996 Eric Freeman, Ph.D. (1996) in Computer Science at Yale University. Advisor is David Gelernter. “Lifestreams”, a computer interface that eliminates the need for filenames by indexing all files and connecting applications to a database.

Proposal Committee:

- 2019 Maddalena Romano in Geographic Information Systems (City College of New York) Advisor is Carsten Kessler. “Distributed WiFi for Distributed GIS”
- 2018 Virginia Leigh Byrne in Education. Advisor is Diane Jass Ketelhut. “The Chilling Effect of Cyberbullying on Women’s Social Presence in Online Discussions”
- 2017 Zhenpeng Zhao in HCI. Advisor is Niklas Elmqvist. “A Framework of Media Types for Data-Driven Storytelling”
- 2012 Irene Eleta in iSchool. Advisor is Jen Golbeck. “Multilingual Use of Twitter: Language Choice and Language Bridges in a Social Network”
- 2012 John Alexis Guerra Gómez in Computer Science. Advisor is Ben Shneiderman. “Exploring Differences in Trees by Comparing Structure and Node Value Changes: An Interactive Information Visualization Approach”
- 2011 Alex Quinn in Computer Science. Advisor is me. “Crowdsourcing Decision Support: Frugal Human Computation for Efficient Decision Input Acquisition”
- 2011 Sureyya Tarkan in Computer Science. Advisor is Ben Shneiderman. “Enhancing Timely Management of Workflow with Interactive Visual Displays and Actions for Situation Awareness Processing”
- 2010 Chris Hayden in Computer Science. Advisors are Mike Hicks and Jeff Foster. “Clear and Correct Runtime Upgrades”
- 2010 Chang Hu in Computer Science. Advisor is me. “Translation by Collaboration Between Monolinguals”
- 2008 Walky Rivadeneira in Psychology. Advisor is Kent Norman. “Tag Clouds: How Format and Categorical Structure Affect Categorization Judgement”
- 2007 Jerry Fails in Computer Science. Advisor is Allison Druin. Research on children’s collaborative mobile interfaces.
- 2005 Haibin Ling in Computer Science. Advisor is David Jacobs. Research on invariance in computer vision.
- 2001 Jaime Montemayor in Computer Science. Advisor is Allison Druin. Research on physical programming for children.
- 2001 Harry Hochheiser in Computer Science. Advisor is Ben Shneiderman. Research on interfaces for time-series data.

2001	Angela Boltman in Education. Advisor is Allison Druin. Research on the technology of storytelling.
2000	Jeff Heflin in Computer Science. Advisor is Jim Hendler. Research on knowledge representation on the Web.
1998	Egemen Tanin in Computer Science. Advisor is Ben Shneiderman. Research on query previews.
1998	Thanarat Horprasert in Computer Science. Advisor is Larry Davis. Research on real-time computer vision.
1998	Vasanth Philomin in Computer Science. Advisor is Larry Davis. Research on real-time computer vision.
1998	Chris North in Computer Science. Advisor is Ben Shneiderman. Research on Tightly Coupled Windows.

iv. Post-doc

2009-2012	Tom Yeh working in applied computer vision and HCI (co-mentored with Larry Davis). Currently at Univ. of Colorado, Boulder.
-----------	---

4. SERVICE

A. PROFESSIONAL

i. Office and Committee Memberships

2015 – 2018	SIGCHI Adjunct Chair for Awards
2015	TOCHI Special Issue Editor
2015	Learning @ Scale Program Committee
2014	Learning @ Scale Program Committee
2011	CHI 2011 Papers Subcommittee Co-chair
2010	CHI 2010 Papers Subcommittee Co-chair
2005 – 2016	US ACM Committee member
2004 – 2007	SIGCHI Adjunct Chair for US Public Policy Committee
2000	CHI Conference Video Paper Co-Chair
1999	CHI Conference Video Paper Chair
1998	UIST Technical Program Committee
1998	UIST Conference Panel Chair, Technical Program Committee
1998	CHI Conference Video Co-Chair.

ii. Editorial Boards

2015 – 2018	PeerJ Computer Science , PeerJ
2012 – 2018	Transactions on Computer-Human Interaction (TOCHI) , ACM Press
2007 – 2014	Foundations and Trends in HCI (Editor-in-chief), NOW Publishers
2002 – 2016	Information Visualization , Palgrave Press

iii. Advisory Boards

2016 – 2018	Coursera , Mountain View, CA Coursera Council advisory group
2012 – 2015	Barrie School , Silver Spring, MD Board of Trustees for this independent K-12 school
2013 – 2016:	Secretary

2003	Microsoft Research , Redmond, WA University Relations Faculty Advisory Board
------	--

iv. Other non-University committees, commissions, panels, etc.

2012	Judge for Joan Ganz Cooney, National STEM Video Game Challenge
------	---

vi. Paid Consultancies

2022 – present	IT Park University Honorary Rector to help lead new tech educational institution	Tashkent, Uzbekistan
2022 – present	EPAM Chief Academic Advisor - help with corporate educational programs and creation of new universities	Newtown, PA
2020 – 2022	Logitech Consultant to help with new product development	Seattle, WA
2009 – present	Consulting as an expert witness largely for patent infringement cases for a range of clients	Windsor, MA
2006 – 2014	Zumobi, Inc. Founder of startup company to create mobile information services software	Seattle, WA
2008 – 2009	Microsoft Research Consultant to help with products related to my research	Seattle, WA
2005 – 2009	Hillcrest Laboratories, Inc. Advisor to help this start-up develop novel in-home entertainment system	Rockville, MD
2003 – 2008	Windsor Interfaces, Inc. President and CEO of start-up company to commercialize some HCIL projects	Chevy Chase, MD
2005	ScholarOne, Inc. Consultant to help them with visualization in their scholarly publication and conference management software	Charlottesville, VA
2004 – 2005	ATTAP, Inc. Consultant to help this start-up develop novel mobile device interfaces for personal information management	New York, NY
2003	Microsoft Research Corp On University Relations Faculty Advisory Board	Redmond, WA
2001 – 2003	PARC Advised on project to support brainstorming	Palo Alto, CA
2000	NASA Goddard Space Flight Center Designed prototype visualizations for satellite constellation control	Greenbelt, MD
1999 – 2001	Swedish Institute of Computer Science (SICS) Worked on European Union KidStory project	Stockholm, Sweden
1998 – 1999	Lockheed Martin Astronautics Advised on use of Pad++ for image analyst application	Denver, CO
1996 – 1997	Sony Research Corporation Advised on use of Pad++ for product development	New York, NY

B. UNIVERSITY

i. Departmental

2013 – 2014	CS Educational Transformation Committee co-chair
2008 – 2012	CS Teaching Evaluation Committee
2008 – 2011	CS Department Council
2007 – 2010	Software Engineering Field Committee chair
2010	UMIACS APT Committee
2009	CS Faculty Search Committee
2007	Chair of APT committee for Evan Golub's promotion to Senior Lecturer
2006	Undergraduate curriculum committee
2006	CS Distinguished Lecturer committee
2004	UMIACS steering committee
2004	High school programming contest
2004	Undergraduate curriculum
2003 – 2005	CS Department Council
2002	CFAR faculty recruiting committee
2001 – 2002	Lab committee
2001 – 2002	On department faculty recruiting committee
2001	Taught 1.5 hour course for high school students in Governor's program.
2000	Represented the computer science department on Capitol Hill for an exhibition to encourage Congressional appropriations for NSF.
2000	Helped with Computer Science Dept. research review day – made departmental demo.
1999 – 2000	Technical staff liaison – to help CS admin and technical staff.
1999	Helped with Computer Science Programming Contest
1996 – 1997	Graduate Admissions coordinator for the UNM Computer Science Department.

iii. University

2016 – 2018	USM Faculty Workload Workgroup
2015 – 2018	DIT Learning Technology Working Group
2015 – 2017	Senate Library Council Committee
2015 – 2018	Senate Educational Affairs Committee
2014 – 2018	Provost's Data Policy Advisory Committee
2013 – 2018	USM Academic Transformation Advisory Committee
2013 – 2018	Future of Information Alliance brainstorming board
2013 – 2018	DIT Technology Accessibility Committee
2013 – 2014	Journalism Faculty Search Committee
2013 – 2014	Learning @ Technology Working Group for DIT
2012 – 2013	Provost's Commission on Blended and Online Education

2012 – 2013	VP IT/CIO – IT Council
2012	CMNS Distinguished Research Scientist Prize committee
2011	VP / CIO Search Committee
2010	UMD Campus-Wide Mobile App Programming Contest Chair
2002 – 2005	Advisory board member for MITH (Maryland Institute for Technology and the Humanities)
2000	Served on search committee for new intellectual property manager for Office of Technology Commercialization.
2000	Ran event for Maryland Day – opened lab to children to use KidPad to tell collaborative stories

APPENDIX 2: MATERIALS CONSIDERED IN THE PREPARATION OF THIS DECLARATION

Exhibit No.	Description
1001	U.S. Patent No. 8,918,741 to Lindberg et al (“the ’741 Patent”)
1002	Prosecution History for the ’741 Patent
1004	U.S. Patent Application Publication No. 2007/0180492 (“Hassan”)
1005	U.S. Patent Application Publication No. 2006/0246955 (“Nirhamo”)
1006	U.S. Patent No. 7,689,939 (“Becker”)
1007	U.S. Patent Application Publication No. 2008/0222545 (“Lemay”)
1008	<i>Nokia Corp. v. Amazon.com, Inc.</i> , No. 1:23-cv-01232 (D. Del.) First Amended Complaint
1009	<i>Nokia Corp. v. Amazon.com, Inc.</i> , No. 1:23-cv-01232 (D. Del.) Complaint - Exhibit 23
1010	European Patent No. 0664020B1 (“Matheny”)
1011	Macintosh Human Interface Guidelines, Apple, 1992
1012	The New Oxford American Dictionary, 2nd edition, 2005
1013	Webster’s College Dictionary, 2005

APPENDIX 3: CHALLENGED CLAIMS

[1a]. An apparatus comprising:

at least one processor; and

at least one memory storing computer program code configured, working with the at least one processor, to cause at least the following to be performed:

[1b] causing a display of a device to display an object, relating to an application, when the device is in a locked mode; and

[1c] enabling a user to select the object while the device is in the locked mode, in order to determine a state, function or utility of the application without fully activating the application while the device remains in the locked mode.

2. The apparatus as claimed in claim 1, wherein the computer program code is configured, working with the at least one processor, to cause at least the following to be performed: responding to user selection of the object by illuminating or highlighting the object.

3. The apparatus as claimed in claim 2, wherein a movable object is provided to enable the user to select the object.

4. The apparatus as claimed in claim 2, wherein the object is highlighted or illuminated to a lesser extent than that associated with a standard mode setting.

5. The apparatus as claimed in claim 1, wherein the object represents an application that is present or available in the device.

6. The apparatus as claimed in claim 1, wherein the computer program code is configured, working with the at least one processor, to cause at least the following to be performed: causing the display to display another object, relating to another application, when the device is in a locked mode.

7. The apparatus as claimed in claim 6, wherein the another object represents an application that is present or available in the device.

8. The apparatus as claimed in claim 6, wherein the another object indicates that the another application is not activatable when the device is in the locked mode.

9. The apparatus as claimed in claim 1, wherein the display is a touch screen display, and enabling a user to select the object comprises enabling the user to select the object using the touch screen display.

10. The apparatus as claimed in claim 1, wherein the apparatus is the device.

[11a]. A method, comprising:

[11b] causing a display of a device to display an object, relating to an application, when the device is in a locked mode; and

[11c] enabling a user to select the object while the device is in the locked mode, in order to determine a state, function or utility of the application without fully activating the application while the device remains in the locked mode.

12. The method as claimed in claim 11, wherein the object is a dimmed icon.

13. The method as claimed in claim 11, further comprising: responding to user selection of the object by illuminating or highlighting the object.

14. The method as claimed in claim 11, wherein a movable object is provided to enable the user to select the object.

15. The method as claimed in claim 11, wherein the object represents an application that is present or available in the device.

16. The method as claimed in claim 11, wherein the display is a touch screen display, and enabling a user to select the object comprises enabling the user to select the object using the touch screen display.

[17a] A non-transitory computer readable medium storing computer program code that, when performed by at least one processor, causes at least the following to be performed:

[17b] causing a display of a device to display an object, relating to an application, when the device is in a locked mode; and

[17c] enabling a user to select the object while the device is in the locked mode, in order to determine a state, function or utility of the application without fully activating the application while the device remains in the locked mode.

18. The non-transitory computer readable medium as claimed in claim 17, wherein the object is a dimmed icon.

19. The apparatus as claimed in claim 17, wherein the object is an icon.

20. The apparatus as claimed in claim 19, wherein the icon is a dimmed icon.

APPENDIX 4: UNDERSTANDING OF THE LAW

I have applied the following legal principles provided to me by counsel in arriving at the opinions set forth in this report.

Legal Standard for Prior Art

I am not an attorney. I have been informed by attorneys of the relevant legal principles and have applied them to arrive at the opinions set forth in this declaration.

I understand that the petitioner for inter partes review may request the cancelation of one or more claims of a patent based on grounds available under 35 U.S.C. § 102 and 35 U.S.C. § 103 using prior art that consists of patents and printed publications.

Anticipation and Prior Art

I understand that § 102 specifies when a challenged claim is invalid for lacking novelty over the prior art, and that this concept is also known as “anticipation.” I understand that a prior art reference anticipates a challenged claim, and thus renders it invalid by anticipation, if all elements of the challenged claim are disclosed in the prior art reference. I understand the disclosure in the prior art reference can be either explicit or inherent, meaning it is necessarily present or implied. I understand that the prior art reference does not have to use the same words as the challenged claim, but all of the requirements of the claim must

be disclosed so that a person of ordinary skill in the art could make and use the claimed subject-matter.

In addition, I understand that § 102 also defines what is available for use as a prior art reference to a challenged claim.

Under § 102(a), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country before the challenged claim's date of invention.

Under § 102(b), a challenged claim is anticipated if it was patented or described in a printed publication in the United States or a foreign country more than one year prior to the challenged patent's filing date.

Under § 102(e), a challenged claim is anticipated if it was described in a published patent application that was filed by another in the United States before the challenged claim's date of invention, or was described in a patent granted to another that was filed in the United States before the challenged claim's date of invention.

I understand that a challenged claim's date of invention is presumed to be the challenged patent's filing date. I also understand that the patent owner may establish an earlier invention date and "swear behind" prior art defined by § 102(a) or § 102(e) by proving (with corroborated evidence) the actual date on which the

named inventors conceived of the subject matter of the challenged claim and proving that the inventors were diligent in reducing the subject matter to practice.

I understand that the filing date of a patent is generally the filing date of the application filed in the United States that issued as the patent. However, I understand that a patent may be granted an earlier effective filing date if the patent owner properly claimed priority to an earlier patent application.

I understand that when a challenged claim covers several structures, either generically or as alternatives, the claim is deemed anticipated if any of the structures within the scope of the claim is found in the prior art reference.

I understand that when a challenged claim requires selection of an element from a list of alternatives, the prior art teaches the element if one of the alternatives is taught by the prior art.

Legal Standard for Obviousness

I understand that even if a challenged claim is not anticipated, it is still invalid if the differences between the claimed subject matter and the prior art are such that the claimed subject matter would have been obvious to a person of ordinary skill in the pertinent art at the time the alleged invention.

I understand that obviousness must be determined with respect to the challenged claim as a whole.

I understand that one cannot rely on hindsight in deciding whether a claim is obvious.

I also understand that an obviousness analysis includes the consideration of factors such as (1) the scope and content of the prior art, (2) the differences between the prior art and the challenged claim, (3) the level of ordinary skill in the pertinent art, and (4) “secondary” or “objective” evidence of non-obviousness.

Secondary or objective evidence of non-obviousness includes evidence of: (1) a long felt but unmet need in the prior art that was satisfied by the claimed invention; (2) commercial success or the lack of commercial success of the claimed invention; (3) unexpected results achieved by the claimed invention; (4) praise of the claimed invention by others skilled in the art; (5) taking of licenses under the patent by others; (6) deliberate copying of the claimed invention; and (7) contemporaneous and independent invention by others. However, I understand that there must be a relationship between any secondary evidence of non-obviousness and the claimed invention.

I understand that a challenged claim can be invalid for obviousness over a combination of prior art references if a reason existed (at the time of the alleged invention) that would have prompted a person of ordinary skill in the art to combine elements of the prior art in the manner required by the challenged claim. I understand that this requirement is also referred to as a “motivation to combine,”

“suggestion to combine,” or “reason to combine,” and that there are several rationales that meet this requirement.

I understand that the prior art references themselves may provide a motivation to combine, but other times simple common sense can link two or more prior art references. I further understand that obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may come from market forces.

I understand obviousness to include, for instance, scenarios where known techniques are simply applied to other devices, systems, or processes to improve them in an expected or known way. I also understand that practical and common-sense considerations should be applied in a proper obviousness analysis. For instance, familiar items may have obvious uses beyond their primary purposes.

I understand that the combination of familiar elements according to known methods is obvious when it yields predictable results. For instance, obviousness bars patentability of a predictable variation of a technique even if the technique originated in another field of endeavor. This is because design incentives and other market forces can prompt variations of it, and predictable variations are not the product of innovation, but rather ordinary skill and common sense.

I understand that a particular combination may be obvious if it was obvious to try the combination. For example, when there is a design need or market

pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. This would result in something obvious because the result is the product not of innovation but of ordinary skill and common sense. However, I understand that it may not be obvious to try a combination when it involves unpredictable technologies.

It is further my understanding that a proper obviousness analysis focuses on what was known or obvious to a person of ordinary skill in the art, not just the patentee. Accordingly, I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

It is my understanding that the Manual of Patent Examining Procedure §2143 sets forth the following as exemplary rationales that support a conclusion of obviousness:

Combining prior art elements according to known methods to yield predictable results;

Simple substitution of one known element for another to obtain predictable results;

Use of known technique to improve similar devices (methods, or products) in the same way;

Applying a known technique to a known device (method, or product) ready for improvement to yield predictable results;

Choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success;

Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art;

Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

A person of ordinary skill in the art looking to overcome a problem will often use the teachings of multiple publications together like pieces of a puzzle, even though the prior art does not necessarily fit perfectly together. Therefore, I understand that references for obviousness need not fit perfectly together like puzzle pieces. Instead, I understand that obviousness analysis takes into account inferences, creative steps, common sense, and practical logic and applications that a person of ordinary skill in the art would employ under the circumstances.

I understand that a claim can be obvious in light of a single reference, if the elements of the challenged claim that are not explicitly or inherently disclosed in the reference can be supplied by the common sense of one of skill in the art.

I understand that obviousness also bars the patentability of applying known or obvious design choices to the prior art. One cannot patent merely substituting one prior art element for another if the substitution can be made with predictable results. Likewise, combining prior art techniques that are interoperable with respect to one another is generally obvious and not patentable.

In order for a claim to be found invalid based upon a modification or combination of the prior art, there must be reasonable expectation that a person of ordinary skill would have successfully modified or combined the prior art to arrive at the claimed arrangement. This does not mean that it must be certain that a person of ordinary skill would have been successful – the law only requires that the person of ordinary skill in the art would have perceived a reasonable expectation of success in modifying or combining the prior art to arrive at the claimed invention.

In sum, my understanding is that obviousness invalidates claims that merely recite combinations of, or obvious variations of, prior art teachings using understanding and knowledge of one of skill in the art at the time and motivated by the general problem facing the inventor at the time. Under this analysis, the prior art references themselves, or any need or problem known in the field of endeavor at the time of the invention, can provide a reason for combining the elements of or attempting obvious variations on prior art references in the claimed manner.

Legal Standard for Claim Construction

I understand that before any invalidity analysis can be properly performed, the scope and meaning of the challenged claims must be determined by claim construction.

I understand that a patent may include two types of claims, independent claims and dependent claims. I understand that an independent claim stands alone and includes only the limitations it recites. I understand that a dependent claim depends from an independent claim or another dependent claim. I understand that a dependent claim includes all the limitations that it recites in addition to the limitations recited in the claim (or claims) from which it depends.

In comparing the challenged claims to the prior art, I have carefully considered the patent and its file history in light of the understanding of a person of skill at the time of the alleged invention.

I understand that to determine how a person of ordinary skill would have understood a claim term, one should look to sources available at the time of the alleged invention that show what a person of skill in the art would have understood disputed claim language to mean. It is my understanding that this may include what is called “intrinsic” evidence as well as “extrinsic” evidence.

I understand that, in construing a claim term, one should primarily rely on intrinsic patent evidence, which includes the words of the claims themselves, the

remainder of the patent specification, and the prosecution history. I understand that extrinsic evidence, which is evidence external to the patent and the prosecution history, may also be useful in interpreting patent claims when the intrinsic evidence itself is insufficient. I understand that extrinsic evidence may include principles, concepts, terms, and other resources available to those of skill in the art at the time of the invention.

I understand that words or terms should be given their ordinary and accepted meaning unless it appears that the inventors were using them to mean something else or something more specific. I understand that to determine whether a term has special meaning, the claims, the patent specification, and the prosecution history are particularly important, and may show that the inventor gave a term a particular definition or intentionally disclaimed, disavowed, or surrendered claim scope.

I understand that the claims of a patent define the scope of the rights conferred by the patent. I understand that because the claims point out and distinctly claim the subject matter which the inventors regard as their invention, claim construction analysis must begin with and is focused on the claim language itself. I understand that the context of the term within the claim as well as other claims of the patent can inform the meaning of a claim term. For example, because claim terms are normally used consistently throughout the patent, how a term is used in one claim can often inform the meaning of the same term in other claims.

Differences among claims or claim terms can also be a useful guide in understanding the meaning of particular claim terms.

I understand that a claim term should be construed not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the entire specification. I understand that because the specification is a primary basis for construing the claims, a correct construction must align with the specification.

I understand that the prosecution history of the patent as well as art incorporated by reference or otherwise cited during the prosecution history are also highly relevant in construing claim terms. For instance, art cited by or incorporated by reference may indicate how the inventor and others of skill in the art at the time of the invention understood certain terms and concepts. Additionally, the prosecution history may show that the inventors disclaimed or disavowed claim scope, or further explained the meaning of a claim term.

With regard to extrinsic evidence, I understand that all evidence external to the patent and prosecution history, including expert and inventor testimony, dictionaries, and learned treatises, can also be considered. For example, technical dictionaries may indicate how one of skill in the art used or understood the claim terms. However, I understand that extrinsic evidence is considered to be less

reliable than intrinsic evidence, and for that reason is generally given less weight than intrinsic evidence.

I understand that in general, a term or phrase found in the introductory words or preamble of the claim, should be construed as a limitation if it recites essential structure or steps, or is necessary to give meaning to the claim. For instance, I understand preamble language may limit claim scope: (i) if dependence on a preamble phrase for antecedent basis indicates a reliance on both the preamble and claim body to define the claimed invention; (ii) if reference to the preamble is necessary to understand limitations or terms in the claim body; or (iii) if the preamble recites additional structure or steps that the specification identifies as important.

On the other hand, I understand that a preamble term or phrase is not limiting where a challenged claim defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention. I understand that to make this determination, one should review the entire patent to gain an understanding of what the inventors claim they invented and intended to encompass in the claims.

I understand that 35 U.S.C. § 112 ¶ 6 created an exception to the general rule of claim construction called a “means plus function” limitation. These types of terms and limitations should be interpreted to cover only the corresponding

structure described in the specification, and equivalents thereof. I also understand that a limitation is presumed to be a means plus function limitation if (a) the claim limitation uses the phrase “means for”; (b) the “means for” is modified by functional language; and (c) the phrase “means for” is not modified by sufficient structure for achieving the specified function.

I understand that a structure is considered structurally equivalent to the corresponding structure identified in the specification only if the differences between them are insubstantial. For instance, if the structure performs the same function in substantially the same way to achieve substantially the same result. I further understand that a structural equivalent must have been available at the time of the issuance of the claim.