

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.

IPR2024-01043

U.S. Patent No. 8,918,741

PETITION FOR *INTER PARTES* REVIEW

OF U.S. PATENT NO. 8,918,741

TABLE OF CONTENTS

EXHIBIT LIST	4
I. Overview of the Technology	5
II. The '741 patent	6
A. Overview	6
B. District Court Allegations.....	7
C. Priority Date	7
III. Field and Level of Skill	7
IV. Claim Construction.....	8
V. Invalidity Grounds	8
A. Grounds 1 and 2.....	8
1. Hassan.....	9
2. Nirhamo.....	11
3. Independent Claims 1, 11, 17.....	12
4. Dependent Claims 2, 13	27
5. Dependent Claims 3, 14	29
6. Dependent Claim 4.....	31
7. Dependent Claims 5, 15	34
8. Dependent Claim 6.....	35
9. Dependent Claim 7.....	37
10. Dependent Claim 8.....	38
11. Dependent Claims 9, 16.....	40
12. Dependent Claim 10.....	42
13. Dependent Claims 12, 18	42
14. Dependent Claim 19.....	43
15. Dependent Claim 20.....	43
B. Grounds 3 and 4.....	43
1. Becker.....	44
2. Independent Claims 1, 11, 17.....	46
3. Dependent Claims 2, 13	60

4. Dependent Claims 3, 14	62
5. Dependent Claim 4	64
6. Dependent Claims 5, 15	66
7. Dependent Claim 6	67
8. Dependent Claim 7	68
9. Dependent Claim 8	68
10. Dependent Claims 9, 16	69
11. Dependent Claim 10	71
12. Dependent Claims 12, 18	72
13. Dependent Claim 19	72
14. Dependent Claim 20	74
VI. Standing	74
VII. Discretionary Analysis for Review	74
VIII. MANDATORY NOTICES AND FEES	75

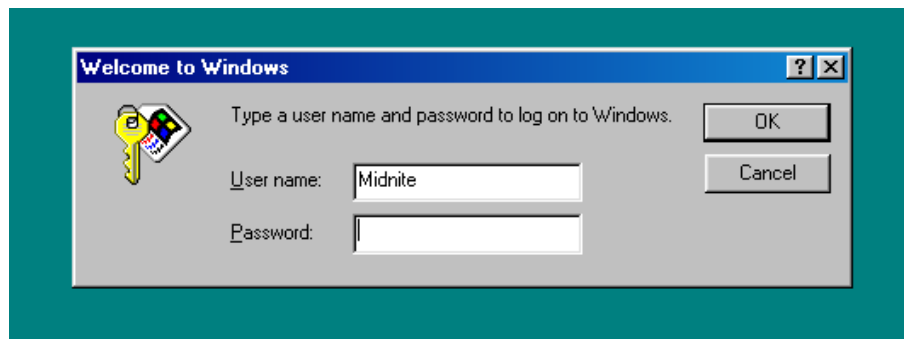
EXHIBIT LIST

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
1003	Declaration of Benjamin B. Bederson, Ph.D.
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

The '741 patent purports to allow limited user access to a locked device. But this was already well-known in the art and taught by Hassan (Grounds 1-2) and Becker (Grounds 3-4) years before the '741 patent.

I. OVERVIEW OF THE TECHNOLOGY

For many decades, going back to Unix and early Windows computers of the 1990s, passwords were required to unlock computers to prevent unauthorized access. Below is an exemplary Windows 98 prompt for unlocking a PC:



By the 2000s, mobile computers, including mobile phones, also included locked and unlocked modes. Bederson, ¶¶3-4.

Several scenarios were recognized in the art where users wanted limited functionality without having to unlock their computers/phones: e.g., someone sharing a device with co-workers to show them a particular email or sharing a device with children for the purpose of gaming. *E.g.*, Ex-1004, ¶¶1-3. Therefore, the prior art included numerous examples of PCs and other devices that allowed limited interaction with certain applications (but not others), without requiring that

the device be unlocked. This included limited access to messaging, email, and PC calling applications while the devices remained locked. *Id.*; Bederson, ¶5.

II. THE '741 PATENT

A. Overview

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, admitting it was known, including, for example, in the form of screen savers. Ex-1001, 1:23-25 (“Background” section), 4:17-19. Bederson, ¶6.

Instead, the '741 patent purportedly allows users to interact with applications, without fully activating them, while the device remains locked. Bederson ¶7. 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; Bederson ¶8.

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

B. District Court Allegations

In district court, 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; Bederson, ¶11.

C. Priority Date

The '741 patent cites a provisional application dated June 29, 2007. This Petition analyzes obviousness as of that date, and all relied-upon art predates June 29, 2007. Bederson, ¶12.

III. FIELD AND LEVEL OF SKILL

The field of the '741 patent is user interfaces. Ex-1001, 1:16-18; Bederson, ¶14. A POSITA at the time of alleged invention of the '741 patent would have had 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. Bederson, ¶¶15-17.

IV. CLAIM CONSTRUCTION

Claims should be construed according to the ordinary and customary meaning as understood by a POSITA.

V. INVALIDITY GROUNDS

Petitioner requests *inter partes* review and cancelation of the Challenged Claims on 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

Bederson, ¶¶20-21.

The Grounds render the challenged claims obvious because any differences between the claimed subject matter and the prior art are such that the subject matter, as a whole, would have been obvious at the time the invention was made to a POSITA to which the subject matter pertains. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007).

A. Grounds 1 and 2

A POSITA would have found the Challenged Claims obvious based on Hassan (Ground 1). Several dependent claims recite simple limitations for, e.g., highlighting or illuminating an object (claims 2, 4), using a movable object (claim 3), using a touchscreen 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. Bederson, ¶22.

Hassan teaches a mobile device that allows limited access to a communication application while the mobile device is in a locked state. *Infra* §V.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 touchscreen in a mobile device. *Infra* §§V.A.6 (explaining application of Nirhamo's dimming teachings to Hassan), V.A.11 (explaining application of Nirhamo's touchscreen teachings to Hassan). These had been standard features in computers for many years before 2007. Therefore, Ground 2 combines known, prior art elements according to known methods to yield predictable results (e.g., a touchscreen that dims objects during a locked state). Bederson, ¶23.

The motivation and reasonable expectation of success in combining these prior art elements is explained in more detail below. *Infra* §§V.A.4, 6, 11; Bederson, ¶24.

1. Hassan

U.S. Patent Application Publication No. 2007/0180492 (Hassan) was filed February 1, 2006, published 2007, and is prior art under pre-AIA §102(e). Bederson, ¶25.

Hassan “enabl[es] sharing of ... 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):

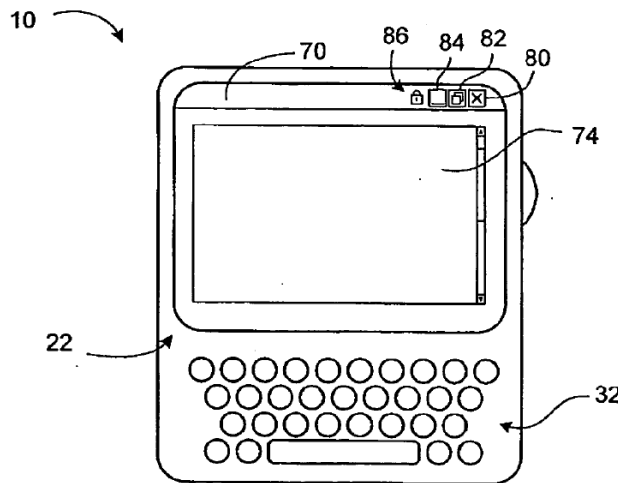


FIG. 3

Bederson, ¶27.

This is useful for allowing others to read particular emails or access particular applications. Accordingly, Hassan displays icons related to applications when the device is in locked state. Ex-1004, ¶59, ¶¶70-72, ¶36. Hassan allows users to select these icons in a locked state to, for example, “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; Bederson, ¶28.

Hassan is in the same field of endeavor as the ’741 patent because it is directed to user interfaces. *See supra* §III; Ex-1004, ¶1; Bederson, ¶29.

2. Nirhamo

U.S. Patent Application Publication No. 2006/0246955 (Nirhamo) was filed May 2, 2005, published November 2, 2006 and is prior art under pre-AIA §102(a), (e). Bederson, ¶31.

Nirhamo “relates to improved presentation, navigation, selection and/or operation options for portable communication devices, particularly, user interface options[.]” Ex-1005, ¶1. Nirhamo discloses device operations, including highlighting items (e.g., an icon or element) when they are selected. Ex-1005, Abstract, ¶4, ¶¶29-31, ¶¶36-¶41, ¶51, ¶71, Fig. 2A; Bederson, ¶32.

Nirhamo teaches dimming items that are unavailable or otherwise incompatible with an operation. Ex-1005, ¶36, ¶¶53-57, ¶60, Figs. 10B-10C, Figs. 11A-11C; Bederson, ¶33.

Nirhamo is in the same field of endeavor as the ’741 patent because it is directed to user interfaces. *See supra* §III; Ex-1005, ¶1; Bederson, ¶34.

3. Independent Claims 1, 11, 17

Beyond their preambles, these claims recite identical limitations. Bederson,

¶35.

[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:**

Hassan teaches **an apparatus** (e.g., a “mobile device”). Ex-1004, ¶¶5-10,

¶20, ¶43, ¶52, ¶68, Fig. 3:

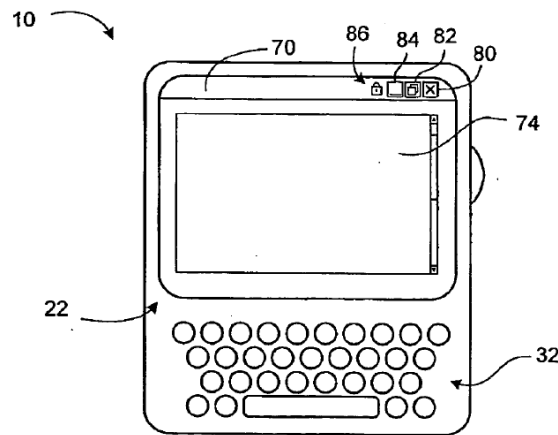


FIG. 3

Bederson, ¶36.

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, ¶15, ¶17, ¶¶26-29, Fig. 1:

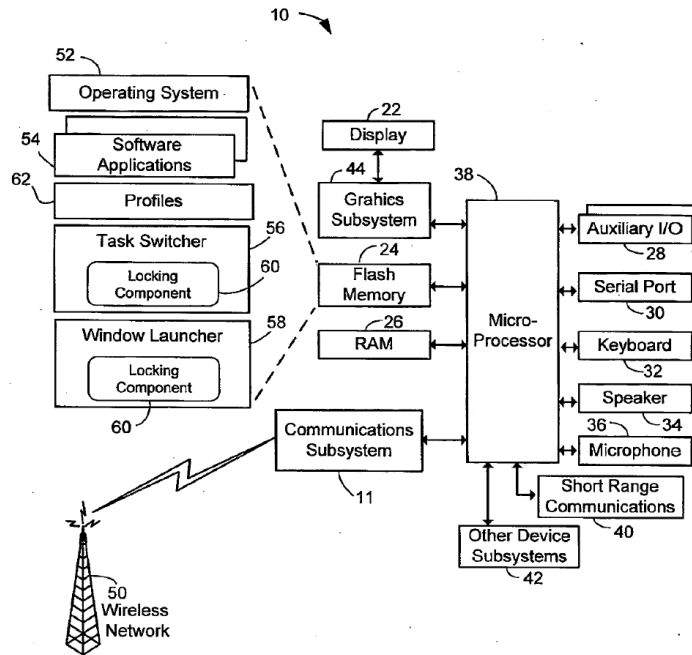


FIG. 1

Therefore, Hassan teaches its mobile device as an apparatus that comprises such a processor and memory. Bederson, ¶37.

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. Bederson, ¶38.

Grounds 1-2 thus teach [1a]. Bederson, ¶39.

[11a]. A method, comprising:

Hassan teaches a **method**, including operations as described for limitations [11b]-[11c] and claims 12-16. 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, ¶¶8-10, ¶¶13-¶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* §§V.A.3[11b]-[11c]; Bederson, ¶40. Grounds 1-2 thus teach [11a]. Bederson, ¶41.

[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:

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 ... 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. Bederson, ¶42. Grounds 1-2 thus teach [17a]. Bederson, ¶43.

[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

Hassan teaches limitations [1b]/[11b]/[17b]. Hassan teaches **a device** (e.g., a mobile device). *Supra* §V.A.3[1a]; Ex-1004, ¶¶5-10, ¶20, ¶43, ¶52, ¶68, Fig. 3; Bederson, ¶44. 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. Bederson, ¶44.

First, Hassan teaches icons¹ 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-

¹ The '741 patent uses “object” broadly and confirms that icons fall within its scope. *E.g.*, '741 patent, 5:15-52, 6:30-55, 7:64-8:18.

1004, ¶59 (“FIG. 5[] shows ... an embodiment of a method[] for managing task change requests *in a locked device*. The method[] begins with the receipt of a task change request in step 122. ... For example, the user may attempt to launch *a new application from a start menu or icon*.”).² 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 for the user to select. *Id.*; Bederson, ¶45.

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,

² All annotations/emphasis added unless otherwise noted.

opening a message, through an appropriate user input (like *clicking on a designated icon*).”), Abstract. 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 GUI for the application. *See id.*; Bederson, ¶46.

Third, Hassan teaches a **locked mode** of the 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. *Id.*, ¶13 (“The user of a locked device may only perform tasks that are permitted without unlocking the device.”), ¶¶14-18, ¶¶38-39 (“[T]he 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. *Id.* As explained below, Hassan allows users to access permitted tasks within an

application when the device is in the locked state. *Infra* §V.A.3[1c]/[11c]/[17c]; Ex-1004, ¶42, ¶38; Bederson, ¶47.

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, verification is required to unlock the device to perform that task: “[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, ¶52, ¶66, Fig. 6; Bederson, ¶48.

The ’741 patent confirms Hassan’s locked state falls within its scope.³ Hassan therefore teaches limitations [1b]/[11b]/[17b]. Grounds 1-2 thus teach limitations [1b]/[11b]/[17b]. Bederson, ¶49.

[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.

Hassan teaches limitations [1c]/[11c]/[17c]. Hassan enables **a user to select the object** (e.g., icon) **while the device** (e.g., mobile device) **is in the locked**

³ For example, like Hassan, the ’741 patent’s embodiments allow limited user interaction to permitted tasks or applications when in locked mode. *See* ’741 patent, 11:32-45, 1:41-53.

mode. As explained above, Hassan teaches two types of icons: (1) for launching an application, and (2) for requesting or initiating operations within an application.

Supra §V.A.3[1b]/[11b]/[17b]. Both satisfy limitations [1c]/[11c]/[17c]. Bederson, ¶50.

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 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. 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 on PCs and mobile devices such as mobile phones. Indeed, Hassan teaches that users initiate task changes by clicking on buttons. *Id.*, ¶¶34-36 (“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; Bederson, ¶51.

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,

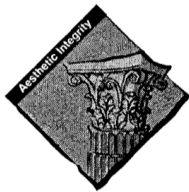
¶¶35-36, Abstract; *supra* §V.A.3[1b]/[11b]/[17b] (explaining how Hassan teaches icons associated with operations). For example, Hassan teaches icons associated with a set of operations (e.g., forwarding, replying to, and opening messages) within an application (e.g., e-mail application) while the device is in locked mode. *Id.* Based on Hassan’s teachings, users select the icon associated with the operation by clicking the icon 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.”); Bederson, ¶52.

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 GUIs of the 1990s

included this functionality. For example, 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 ... People can select, open, move, copy, and throw away icons.” 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

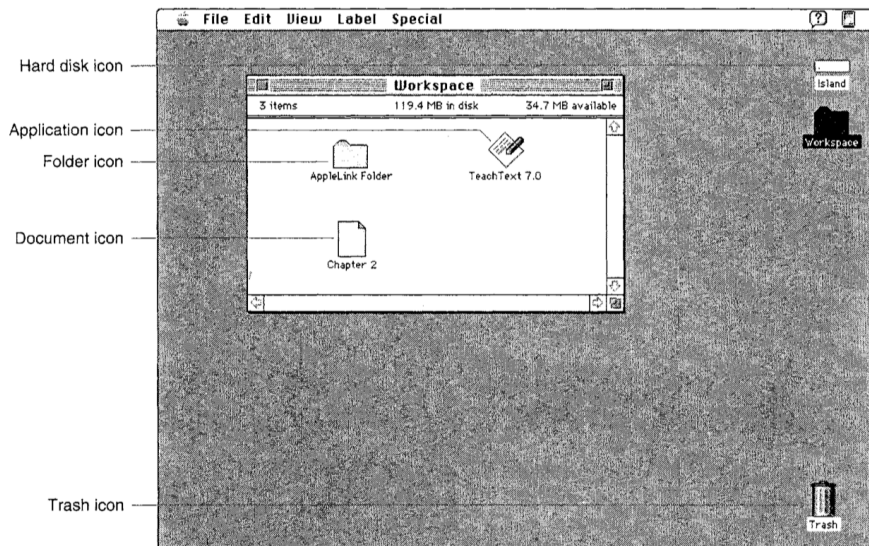
Icons



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, Fig. 8-1. The Apple icon represents a selectable menu option that is part of the Finder application as described in that same document:

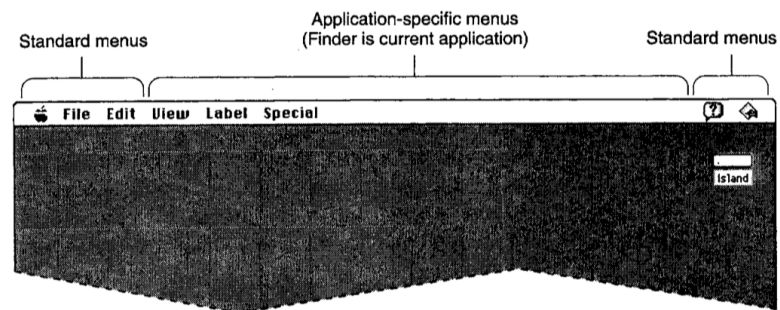
The Menu Bar

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; Bederson, ¶53.

Therefore, by 2007, a POSITA with the given level of education and experience (*supra* §III) 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, Figs. 2-3, ¶36, ¶44, ¶49; Bederson, ¶54.

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. 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; it instead turns to further processes, e.g., displaying an authorization window for password. Ex-1004, ¶61. 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 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 application (e.g., what functions of the application are permitted or not during locked mode), and a utility of the application (e.g., what can the application be used for during locked mode). Hassan therefore 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. Bederson, ¶55.

PO's district court positions confirm that Hassan's teachings fall within the scope of this limitation. For example, PO 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; Bederson, ¶56.

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).

Additionally, Hassan teaches that a user selects such icons in order to determine a state of the application (e.g., whether the associated tasks can be performed in locked mode). As explained above, PO's district court positions confirm that Hassan's teachings fall within the scope of the challenged claims. Bederson, ¶57.

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) 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. Ex-1004, ¶¶70-72, ¶38. Since the application cannot perform its full set of operations, it is not fully activated. *See id.*; Bederson, ¶58.

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 others. 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; Bederson, ¶59. Grounds 1-2 thus teach limitations [1c]/[11c]/[17c]. Bederson, ¶60.

4. Dependent Claims 2, 13

[2/13]. ...

responding to user selection of the object by illuminating or highlighting the object.

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 petition to be performed, and the method of claim 11. *Supra* §V.A.3. Ground 2 further teaches the additional limitations of claims 2 and 13. Bederson, ¶61.

Hassan teaches the user selecting an object related to an application. *Supra* §§V.A.3[1b]/[11b]/[17b]-[1c]/[11c]/[17c]. Nirhamo further teaches **responding to user selection of the object by illuminating or highlighting the object**. Nirhamo relates to “user interface options involved on the display screens[.]” Ex-1005, ¶1. Nirhamo discloses device operations, including highlighting items on the display when selected. Ex-1005, Abstract, ¶4, ¶¶29-31, ¶¶36-41, ¶51, ¶71. For example, Nirhamo teaches that “when a particular icon or element[.] is selected[,] [t]his element[.] may then be highlighted is some way[.]” 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 ... In these embodiments, the ‘items’ 81, 82, 83 may be considered as either selectable items or features[[].”); Bederson, ¶62.

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* §V.A.3[1c]/[11c]/[17c]. 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* §V.A.3[1c]/[11c]/[17c]. 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. 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. Bederson, ¶63.

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 teachings of user selection. Highlighting user selection was implied by Hassan because that is how device interfaces (e.g., computer interfaces) had operated for decades. Ex-1011, 000023-57, Fig. 8-1; Bederson, ¶64.

The combination therefore teaches responding to user selection of the object by illuminating or highlighting the object. Ground 2 thus teaches claims 2 and 13. Bederson, ¶65.

5. Dependent Claims 3, 14

[3/14]. ...

wherein a movable object is provided to enable the user to select the object.

Ground 2 teaches the apparatus of claim 2 and the method of claim 11. (*supra* §§V.A.3, 4) and the additional limitations of claims 2 and 13. Bederson, ¶66.

The combination teaches that a movable object (e.g., cursor) is provided to enable the user to select the object. Hassan teaches that a user selects an object related to an application. *Supra* §V.A.3. Hassan further explains that the user may select the object using a movable object such as a cursor controlled by a mouse.

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 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.*”), ¶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 UI and perform operations such as selecting items. Bederson, ¶67.

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). Ex-1005, ¶39 (“[T]he user can move an emphasized or otherwise highlighted cursor or visual selection representation[] to select the desired functional operation to be performed for the selected item 32[.]”), ¶51, ¶23, ¶38; Bederson, ¶68.

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, selecting, etc., which Hassan already suggests because Hassan

teaches users pointing/clicking using mouses. 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; Ex-1010, 3:22-43, 8:13-26, 16:32-47, 37:51-38:3, 41:8-20, Fig. 1A; Bederson, ¶69.

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

6. Dependent Claim 4

<p>4. ... wherein the object is highlighted or illuminated to a lesser extent than that associated with a standard mode setting.</p>
--

Ground 2 teaches the apparatus of claim 2 (*supra* §V.A.4) and the additional limitations of claim 4. Bederson, ¶71.

The combination teaches responding to user selection of the object by illuminating or highlighting the object. *Supra* §V.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, ¶39; Bederson, ¶72.

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.’ ... 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; Ex-1013, 000003; Bederson, ¶73.

Hassan teaches that some icons are unavailable when the device is locked because the associated applications or the associated operations (within an application) are not permitted during the locked state. *Supra* §V.A.3[1c]/[11c]/[17c]. 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 device 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.

Bederson, ¶74.

Therefore, Hassan teaches that some icons are unavailable during the locked state (*supra* §V.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, ¶¶53-57, ¶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* §V.A.4), and also

dimming and de-emphasizing the item to indicate its unavailability (Ex-1005, ¶36, ¶¶53-57, ¶60, Figs. 10B-10C, Figs. 11A-11C). Ground 2 therefore teaches claim 4. Bederson, ¶¶75-76.

7. Dependent Claims 5, 15

[5/15]. ...

wherein the object represents an application that is present or available in the device.

Hassan teaches the apparatus of claim 1 and method of claim 11 (*supra* §V.A.3) and the additional limitations of claims 5 and 15. Bederson, ¶77.

Hassan displays an object relating to an application (*supra* §§V.A.3[1b]/[11b]/[17b]-[1c]/[11c]/[17c]) and 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 associated icon. *Supra* §V.A.3; Ex-1004, ¶28, ¶59. In addition, the user clicks an icon for an operation to request or initiate that operation within an application on the device. *Supra* §V.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. Bederson, ¶78.

8. Dependent Claim 6

6. ... causing the display to display another object, relating to another application, when the device is in a locked mode.

Hassan teaches the apparatus of claim 1 (*supra* §V.A.3) and the additional limitations of claim 6. Bederson, ¶79.

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**. 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* §V.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):

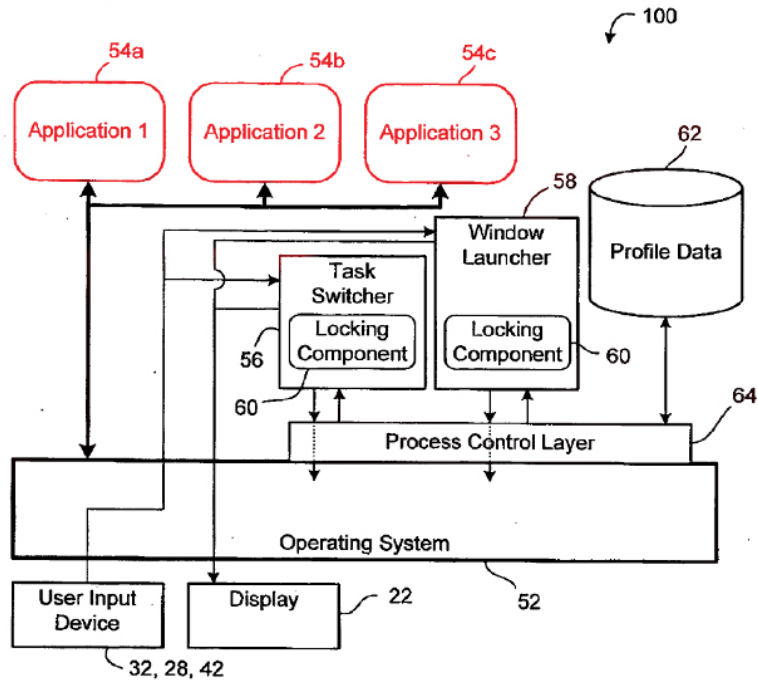


FIG. 2

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; Bederson, ¶80.

Hassan teaches icons for launching applications. Ex-1004, ¶59, ¶¶70-72, ¶36. This was explained above for claim 1. *Supra* §V.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. Bederson, ¶81.

9. Dependent Claim 7

<p>7. ... wherein the another object represents an application that is present or available in the device.</p>

Hassan teaches the apparatus of claim 6 (*supra* §V.A.8) and the additional limitations of claim 7. Bederson, ¶82.

Hassan teaches causing 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* §V.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. Bederson, ¶83.

10. Dependent Claim 8

6. ... causing the display to display another object, relating to another application, when the device is in a locked mode.

Hassan teaches the apparatus of claim 6. *Supra* §V.A.8. Ground 2 further teaches the additional limitations of claim 8. Bederson, ¶84.

Ground 2 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* §V.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; Bederson, ¶85.

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. Bederson, ¶86.

11. Dependent Claims 9, 16

[9/16]. ...

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.

Hassan teaches the apparatus of claim 1 and the method of claim 11. *Supra* §V.A.3. Ground 2 further teaches the additional limitations of claims 9 and 16. For example, Nirhamo teaches that touchscreens could be used to implement the operations it discloses. Ex-1005, ¶23 (“Touch screen functionality could also be used.”). Moreover, using touchscreen displays to facilitate user interactions (e.g., selecting objects on the touchscreen) was well-known by 2007. 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, ...”), ¶119, ¶125 (“In some embodiments, the user interface includes a finish icon ... 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; Bederson, ¶87.

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

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. 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. Bederson, ¶89.

Moreover, the combination would have had several advantages. Touchscreens offered an intuitive UI and were thus often touted as a marketing feature in laptops. Therefore, a POSITA would have had ample motivation to combine Nirhamo's teachings with Hassan. Bederson, ¶90.

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 user interaction 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

touchscreen display, which was common and widely used by 2007. Bederson, ¶91. Ground 2 therefore teaches claims 9 and 16. Bederson, ¶92.

12. Dependent Claim 10

10. ... wherein the apparatus is the device.

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* §V.A.3. Hassan therefore teaches that the apparatus is the device, i.e., the mobile device. Grounds 1-2 therefore teach claim 10. Bederson, ¶93.

13. Dependent Claims 12, 18

[12/18]....

wherein the object is a dimmed icon.

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

14. Dependent Claim 19

19. ... wherein the object is an icon.

Hassan teaches the apparatus of claim 17 (*supra* §V.A.3) where the object is an icon that represents an application or an operation within an application. *Supra* §V.A.3; Ex-1004, ¶59, ¶¶70-72. Hassan therefore teaches that the object is an icon. Bederson, ¶95.

15. Dependent Claim 20

20. ... wherein the icon is a dimmed icon.

Hassan teaches the apparatus of claim 19 where the object is an icon. *Supra* §V.A.14. As explained above, Ground 2 teaches a dimmed icon. *Supra* §V.A.13; Bederson, ¶96.

B. Grounds 3 and 4

A POSITA would have found the Challenged Claims obvious based on Becker (Ground 3). The combination of Becker and Nirhamo (Ground 4) teaches additional limitations for various dependent claims. Bederson, ¶97.

Becker teaches limited user access to a telephone application while the computer is in a locked screen saver mode. *Infra* §V.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 touchscreen display. *Infra* §§V.B.5 (explaining application of Nirhamo's dimming

teachings to Becker), V.B.10 (explaining application of Nirhamo’s touchscreen teachings to Becker). These had been standard features in computers for many years before 2007. Therefore, 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). Bederson, ¶98.

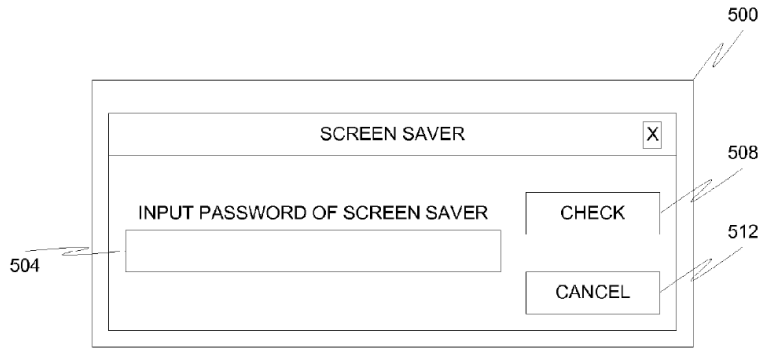
The motivation and reasonable expectation of success in combining these prior art elements is explained in more detail below. *See infra* §§V.B.3, 5, 10; Bederson, ¶99.

1. Becker

U.S. Patent No. 7,689,939 (Becker) was filed as application 10/882,889 on June 30, 2004 and patented on March 30, 2010. Becker is prior art under at least pre-AIA §102(e). Bederson, ¶101.

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[.]” 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, 2:61-3:3, 3:12-29; Bederson, ¶102.

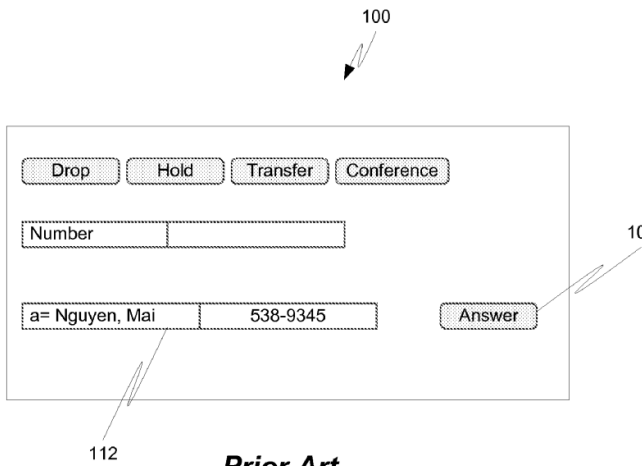
Becker incorporates conventional user interface features into its embodiments and teachings, including “a conventional screen saver display” that needs a password to unlock the computer. Ex-1006, 1:48-2:4, 3:50, Fig. 5:



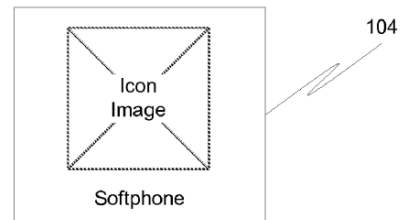
**Prior Art
Fig. 5**

Bederson, ¶103.

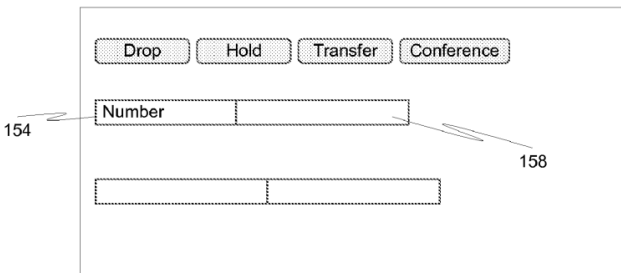
Becker’s teachings also include conventional user interface windows for a phone application, which the user can interact with during screen saver mode. Ex-1006, 2:9-13, 3:41-42, Figs. 1A-1D:



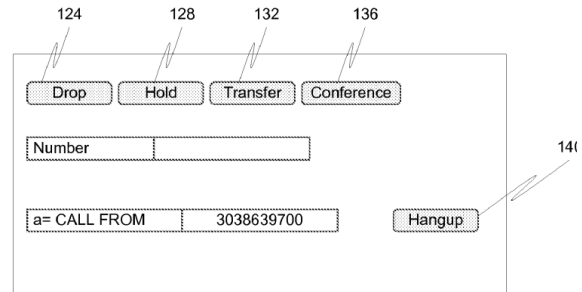
**Prior Art
Fig. 1A**



**Prior Art
Fig. 1B**



**Prior Art
Fig. 1C**



**Prior Art
Fig. 1D**

Bederson, ¶¶104.

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. Additionally, a POSITA would have found it obvious and been motivated to combine Becker's embodiments with Becker's teachings of prior GUI windows because Becker provides express motivation to do so, and it would have had the advantage of using known GUI windows while providing quick access to certain operations.

Bederson, ¶¶106-107.

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

2. Independent Claims 1, 11, 17

Beyond their preambles, these claims recite identical limitations. Bederson, ¶109.

[1a]. An apparatus ...

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, 4:51-67, Fig. 3:

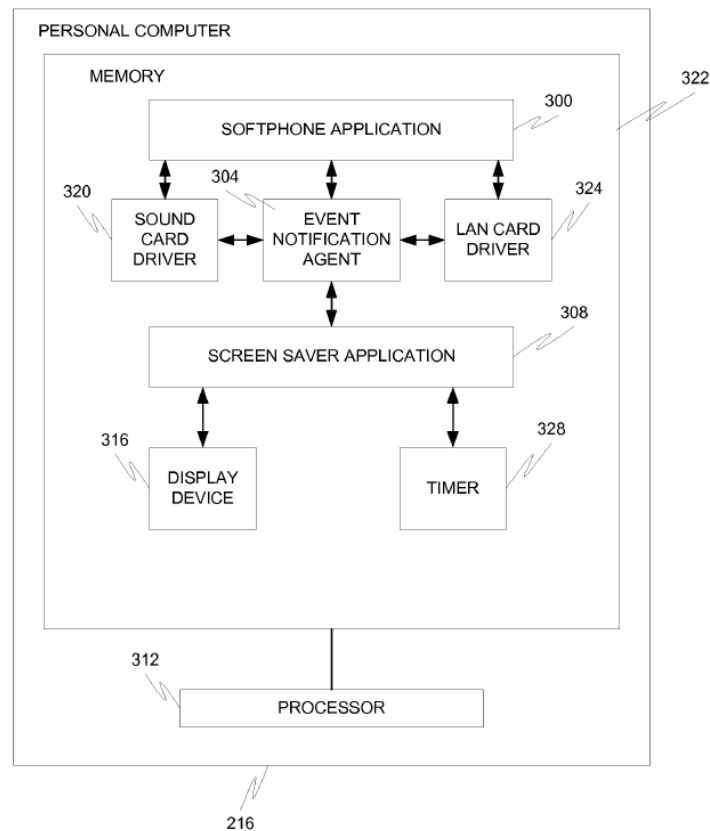


Fig. 3

Therefore, Becker teaches its PC as an apparatus that comprises such a processor and memory. Bederson, ¶110.

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, 4:25-67, 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. Bederson, ¶111.

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

[11a]. A method ...

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* §§V.B.2[11b]-[11c]; Bederson, ¶113. Grounds 3-4 thus teach [11a]. Bederson, ¶114.

[17a]. A non-transitory computer readable medium ...

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. 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. Bederson, ¶115. Grounds 3-4 thus teach [17a]. Bederson, ¶116.

[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

Becker teaches limitations [1b]/[11b]/[17b]. Becker teaches a device (e.g., PC). *Supra* §V.B.2[1a]; Ex-1006, 3:54-64, 4:51-67, Figs. 2-3; Bederson, ¶117.

Becker teaches that the device has a **locked mode** (e.g., screen saver mode). Becker's screen saver mode is a locked mode—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, 6:4-16, 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; Bederson, ¶118.

Becker's screen saver is a locked mode that allows limited interaction with a defined set of applications without fully activating them (*infra* §V.B.2[1c]/[11c]/[17c]) but otherwise prohibits access to normal applications. Ex-1006, 2:40-3:3 (“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.”), 3:12-29, 4:25-40, 6:17-60. The '741 patent confirms that Becker's screen saver mode falls within the scope of the recited “locked mode.”⁴ Bederson, ¶119.

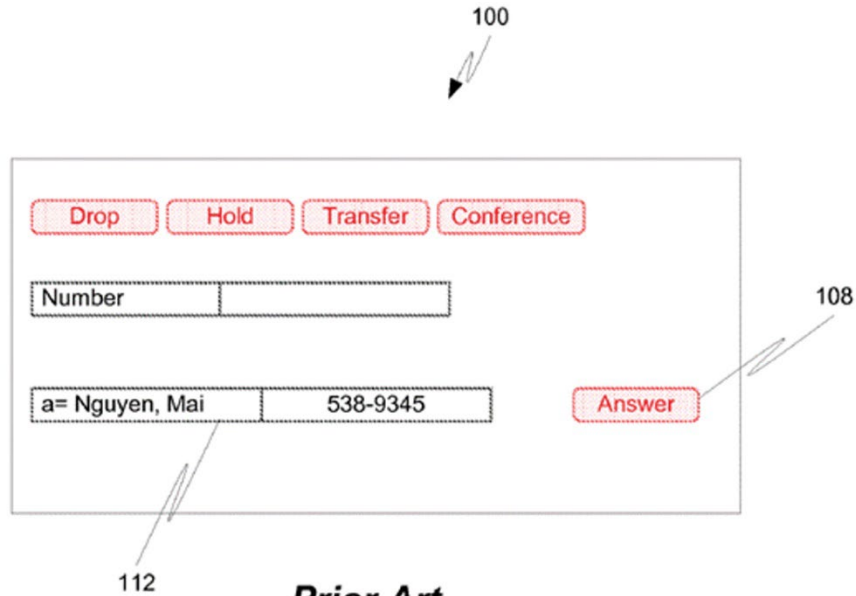
⁴ For example, the '741 patent discusses “screen saver” as locked mode. '741 patent, 4:17-19 (“In alternate embodiments, the icon 206 might move about the

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, 6:17-26, 2:47-3:11. Alternatively, Becker teaches that the GUI display “is always visible on the screen saver display.” Ex-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; Bederson, ¶120.

Becker teaches that the GUI display includes buttons, e.g., “Answer,” “Drop,” “Hold,” “Transfer,” and “Conference” buttons. Ex-1006, 6:48-7:17, Figs. 1A (color added):⁵

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, 1:41-53.

⁵ 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



**Prior Art
Fig. 1A**

Bederson, ¶121.

Each button (e.g., “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 GUI of the

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* §V.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).

application window for the user to click, to perform the corresponding operation.

Id.; Bederson, ¶122.

Becker teaches displaying the application GUI display, including its buttons, **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, 2:42-3:11, Figs. 1A-1C; Bederson, ¶123. Becker therefore teaches limitations [1b]/[11b]/[17b]. Bederson, ¶124.

[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.

Becker teaches limitations [1c]/[11c]/[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, 5:1-24, 6:17-7:17, 2:47-3:11, Fig. 1A; *supra* §V.B.2[1b]/[11b]/[17b]. Becker teaches that users select the GUI display, including, for example, “Answer,” “Drop,” “Hold,” “Transfer,” “Conference” buttons, to perform permitted operations when the PC is in 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 during screen saver mode to answer/drop/hold/transfer/conference an incoming call. *Id.*; Bederson, ¶125.

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.’ ... 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[] 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 id.*; Ex-1005, ¶26, ¶29, Figs. 2-3, ¶¶35-36, ¶44, ¶49; Bederson, ¶126.

Becker further teaches enabling a user to select the object (e.g., button) while the device is in the locked mode **in order to determine a state, function or utility of the application**. Becker enables a user to select its buttons, 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 users selecting the "Answer" button in order to determine the function of the communication application, e.g., whether or not the application supports the functionality of answering incoming calls during screen saver mode. Ex-1006, 2:61-3:11, 6:48-7:17. Likewise, Becker enables users to select the drop/hold/transfer/conference buttons to determine whether or not the application supports the corresponding functionality during screen saver mode. *Id.*; Bederson, ¶127.

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/conference incoming calls) during screen saver mode. Becker's communication application is utilized to answer/drop/hold/transfer/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. Ex-1006, 1:29-47, 5:1-24, 2:61-3:11, 6:48-7:17; Bederson, ¶128.

PO's district court representations confirm that Becker's teachings fall within the scope of this limitation. For example, PO 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. Bederson, ¶129.

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:

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.

Ex-1006, 6:48-60. 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.*; Bederson, ¶130.

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; 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 redisplay the screen saver display."), Fig. 4:

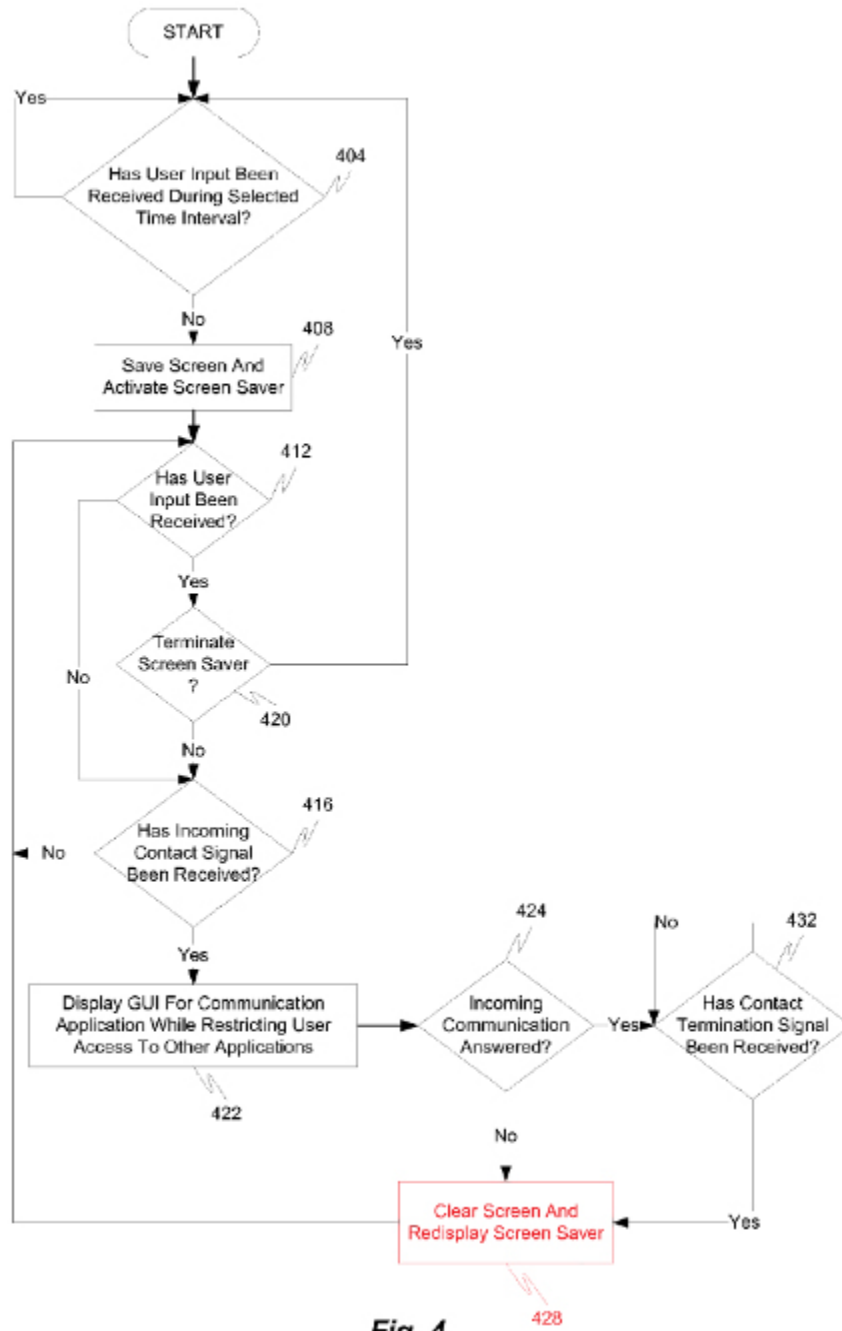


Fig. 4

Bederson, ¶131.

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. Bederson, ¶132. Becker therefore teaches limitations [1c]/[11c]/[17c]. Bederson, ¶133.

3. Dependent Claims 2, 13

[2/13]. ... responding to user selection of the object by illuminating or highlighting the object.

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 petition to be performed, and the method according to claim 11. *Supra* §V.B.2. Ground 4 further teaches the additional limitations of claims 2 and 13. Bederson, ¶134.

Becker teaches the user selecting an object related to an application. *Supra* §§V.B.2[1b]/[11b]/[17b]-[1c]/[11c]/[17c]. Nirhamo further teaches **responding to user selection of the object by illuminating or highlighting the object**. 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; *see also supra* §V.A.4. 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; Bederson, ¶135.

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* §V.B.2.[1c]/[11c]/[17c]. It was customary for device interfaces to highlight 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* §V.B.2[1c]/[11c]/[17c]. 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 was 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.

Bederson, ¶136.

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 that 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; Bederson, ¶137.

Ground 4 therefore teaches responding to user selection of the object by illuminating or highlighting the object and thus teaches claims 2 and 13. Bederson, ¶138.

4. Dependent Claims 3, 14

[3/14]. ... wherein a movable object is provided to enable the user to select the object.

Ground 4 teaches the apparatus of claim 2 and the method of claim 11. (*supra* §§V.B.2, 3) and the additional limitations of claims 2 and 13. Bederson, ¶139.

The combination teaches that a movable object (e.g., cursor) is provided to enable the user to select the object. Becker teaches that a user selects an object relating to an application. *Supra* §V.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, ¶23, ¶38. 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. Bederson, ¶140.

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, ¶40; Ex-1010, 3:22-43, 8:13-26, 16:32-47, 37:51-38:3, 41:8-20, Fig. 1A. Bederson, ¶141.

The combination therefore 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. Bederson, ¶142.

5. Dependent Claim 4

4. ... wherein the object is highlighted or illuminated to a lesser extent than that associated with a standard mode setting.

Ground 4 teaches the apparatus of claim 2 (*supra* §V.B.3) and the additional limitations of claim 4. Bederson, ¶143.

As explained above, the combination teaches responding to user selection of the object by illuminating or highlighting the object. *Supra* §V.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**. Nirhamo teaches illuminating or highlighting an item by changing its brightness. Ex-1005, ¶51, ¶43, ¶39; Bederson, ¶144.

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. Bederson, ¶145.

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* §V.B.2[1c]/[11c]/[17c]; 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. Bederson, ¶146.

Therefore, Becker teaches that some buttons are unavailable during the screen saver mode (*supra* §V.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* §V.B.3), and also dimming and deemphasizing the item to indicate its unavailability (Ex-1005, ¶36, ¶¶53-54, ¶¶56-57, ¶60, Figs. 10B-10C, Figs. 11A-11C). Bederson, ¶147. Ground 4 therefore teaches claim 4. Bederson, ¶148.

6. Dependent Claims 5, 15

[5/15]. ... wherein the object represents an application that is present or available in the device.

Becker teaches the apparatus of claim 1 and the method of claim 11 (*supra* §V.B.2) and the additional limitations of claims 5 and 15. Bederson, ¶149.

Becker displays an object relating to an application.

Supra §§V.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 on the PC 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. Bederson, ¶150.

7. Dependent Claim 6

<p>6. ... causing the display to display another object, relating to another application, when the device is in a locked mode.</p>

Becker teaches the apparatus of claim 1 (*supra* §V.B.2) and the additional limitations of claim 6. Bederson, ¶151.

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.

Bederson, ¶152. Grounds 3-4 therefore teach claim 6. Bederson, ¶153.

8. Dependent Claim 7

7. ... wherein the another object represents an application that is present or available in the device.

Becker teaches the apparatus of claim 6 (*supra* §V.B.7) and the additional limitations of claim 7. Bederson, ¶154.

Becker teaches causing the display to display another object relating to another application, when the device is in a locked mode. *Supra* §V.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. Bederson, ¶155.

9. Dependent Claim 8

8. ... wherein the another object indicates that the another application is not activatable when the device is in the locked mode.

Becker teaches the apparatus of claim 6 (*supra* §V.B.7) and the additional limitations of claim 8. Bederson, ¶156.

Ground 4 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* §V.B.5; Ex-1005, ¶¶53-57 (“Items in the list 86, see particularly items 81 *a*, 82 *a*, and 83 *a*, are de-emphasized or dimmed ... 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.*”), ¶60, Figs. 10B-10C, Figs. 11A-11C; Bederson, ¶157.

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. 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. Bederson, ¶158.

10. Dependent Claims 9, 16

[9/16]. ... 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.

Becker teaches the apparatus of claim 1 and the method of claim 11. *Supra* §V.B.2. Ground 4 further teaches the additional limitations of claims 9 and 16. For

example, Nirhamo teaches that touchscreens 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 well before 2007. Ex-1007, Abstract, ¶¶7-11, ¶77, ¶119, ¶¶125-126, Fig. 2; Ex-1010, 3:22-43; *see also supra* §V.A.11; Bederson, ¶159.

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

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. Bederson, ¶161.

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. Bederson, ¶162.

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 touchscreen display, and touchscreen display was common and widely used by 2007. Bederson, ¶163. Grounds 3-4 therefore teach claims 9 and 16. Bederson, ¶164.

11. Dependent Claim 10

10. ... wherein the apparatus is the device.

Becker teaches the apparatus of claim 1. *Supra* §V.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* §V.B.2. Becker therefore teaches that the apparatus is the device, i.e., the PC. Grounds 3-4 therefore teach claim 10. Bederson, ¶165.

12. Dependent Claims 12, 18

[12/18]. ... wherein the object is a dimmed icon.

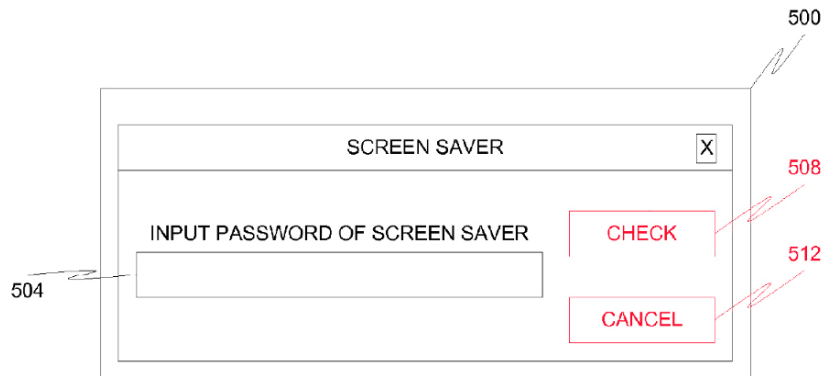
Becker teaches the method of claim 11 and the non-transitory computer readable medium of claim 17. *Supra* §V.B.2. Ground 4 teaches that **the object is a dimmed icon**. Becker teaches that some buttons are unavailable during the locked mode because the associated operations are not permitted (*supra*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

13. Dependent Claim 19

19. ... wherein the object is an icon.

Becker teaches the apparatus of claim 17 (*supra* §V.B.2) where the object (e.g., button in a GUI) related to a communication application. *Supra* §V.B.2; 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

Bederson, ¶167.

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 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.” Ex-1011, 000026. This is also confirmed by Nirhamo. Ex-1005, ¶39 (“In [] FIG. 3 which shows involvement with a music or MP3 player application, the functional icons ... may represent a music play/pause

button 38 *a*, fast forward button 38 *b*”), Figs .3A-3B; Bederson, ¶168. Becker therefore teaches that the object is an icon. Bederson, ¶169.

14. Dependent Claim 20

20. ... wherein the icon is a dimmed icon.

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

VI. STANDING

Petitioner certifies that Petitioner is not barred or estopped from requesting this *inter partes* review, and the '741 patent is IPR-eligible.

VII. DISCRETIONARY ANALYSIS FOR REVIEW

This Petition’s merits are compelling, which “alone demonstrates that the PTAB should not discretionarily deny institution under *Fintiv*.” June 21, 2022 Director’s Memorandum, 3-5.

Fintiv. No denial is warranted under *Fintiv*. The relevant *Fintiv* factors weigh heavily against denial because the lone concurrent district court proceeding was initiated late 2023. No schedule has been entered. Any trial would therefore be

expected well after the FWD deadline (likely November 2025). It is unclear whether any overlap exists between this Petition and future issues in district court.

Advanced Bionics. Denial is likewise not warranted under §325(d) because none of the prior art was cited or considered during prosecution. This is the only IPR filed against the challenged claims to date. *General Plastic* factors do not weigh against institution.

VIII. MANDATORY NOTICES AND FEES

1. **Real Party In Interest**: The real party-in-interest here are Amazon.com, Inc. and Amazon.com Services LLC. No other parties directed, controlled, or funded this *Inter Partes* Review proceeding (IPR).
2. **Related Matters**: Nokia Corporation and Nokia Technologies Oy v. Amazon.com, Inc. and Amazon.com Services, LLC., 1-23-cv-01232 (D. Del.)
3. **Lead and Backup Counsel**: Pursuant to 37 C.F.R. §42.8(b)(3) and 42.10(a), Petitioner designates Harper Batts (Reg. No. 56,160) as lead counsel and Chris Ponder (Reg. No. 77,167) and Jeffrey Liang (Reg. No. 69,043) as back-up counsel, each of Sheppard, Mullin, Richter & Hampton LLP.
4. **Service Information**: Petitioner consents to service by electronic mail addressed to:

HBatts@sheppardmullin.com;
CPonder@sheppardmullin.com;
JLiang@sheppardmullin.com; and
LegalTM-Amazon-Nokia-IPRs@sheppardmullin.com

5. Petitioner's counsel may also be served by mail or hand delivery at Sheppard, Mullin, Richter & Hampton LLP, 1540 El Camino Real Suite 120, Menlo Park, CA 94025. Petitioner's counsel may be reached by telephone at (650) 815-2673 and by facsimile at (650) 815-4668.

6. Fees: The Office is authorized to charge fees for this Petition to Deposit Account 50-4561, Ref. 19AG-382973.

Respectfully submitted,

SHEPPARD, MULLIN,
RICHTER & HAMPTON LLP

/DRAFT/

Harper Batts (Reg. No. 56,160)
Attorneys for Petitioner

CLAIM APPENDIX

[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.

CERTIFICATE OF SERVICE

The undersigned hereby certifies that on June 25, 2024, true and correct copies of the foregoing Petition for *Inter Partes* Review of U.S. Patent No. 8,918,741 and Exhibits 1001-1013 were served in entirety on the Patent Owner at the following addresses of record as listed on PAIR, via Priority Mail Express®:

Nokia Corporation and Alston & Bird LLP
Vantage South End
1120 South Tryon Street, Suite 300
Charlotte, NC 28203-6818

The undersigned certifies that true and correct copies of the Petition for *Inter Partes* Review of U.S. Patent No. 8,918,741 and Exhibits 1001-1013 were also sent via electronic mail to the attorneys of record for Plaintiff in the concurrent litigation matter(s):

Brian E. Farnan
bfarnan@farnanlaw.com
Alexandra F. Easley
aeasley@mckoolsmith.com
John D Haynes
john.haynes@alston.com
Joshua Newcomer
jnewcomer@mckoolsmith.com
Kevin Burgess
kburgess@mckoolsmith.com
Michael J. Farnan
mfarnan@farnanlaw.com

Nicholas T. Tsui
nick.tsui@alston.com
R. Mitch Verboncoeur
mverboncoeur@mckoolsmith.com
Shawn P. Gannon
shawn.gannon@alston.com
Stephen R. Lareau
stephen.lareau@alston.com
Theodore Stevenson, III
ted.stevenson@alston.com
Warren H. Lipschitz
wlipschitz@mckoolsmith.com

SHEPPARD, MULLIN,
RICHTER & HAMPTON LLP

/Harper Batts/
Harper Batts (Reg. No. 56,160)
Attorneys for Petitioner

Date: June 25, 2024

1540 El Camino Real Suite 120

Menlo Park, CA 94025

Tel: (650) 815-2673

**CERTIFICATE OF COMPLIANCE WITH
TYPE-VOLUME LIMITATION, TYPEFACE REQUIREMENTS,
AND TYPE STYLE REQUIREMENTS**

1. This Petition complies with the type-volume limitation of 14,000 words, comprising 13,640 words, as counted using the Microsoft Word software that was used to prepare this paper, excluding the parts exempted by 37 C.F.R. § 42.24(a).

2. This Petition complies with the general format requirements of 37 C.F.R. § 42.6(a) and has been prepared using Microsoft® Word in 14-point Times New Roman.

Respectfully submitted,

SHEPPARD, MULLIN,
RICHTER & HAMPTON LLP

/Harper Batts/
Harper Batts (Reg. No. 56,160)
Attorney for Petitioner

Date: June 25, 2024

1540 El Camino Real Suite 120
Menlo Park, CA 94025
Tel: (650) 815-2673