

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte SCOTT V. TOTMAN,
JEFFREY DAVID KIMBALL, STEVE CLARK,
JAMES ANDREW CANFIELD, ELISA NADER,
THOMAS BRENT CANFIELD, and KEITH R. DEAVEN

Appeal 2008-0690
Application 10/112,029
Technology Center 2100

Decided: March 17, 2008

Before LANCE LEONARD BARRY, HOWARD B. BLANKENSHIP, and
STEPHEN C. SIU, *Administrative Patent Judges*.

BLANKENSHIP, *Administrative Patent Judge*.

DECISION ON APPEAL

This is an appeal under 35 U.S.C. § 134(a) from the Examiner's final rejection of claims 1-15, 17-19, 22-41, 43-45, 48-53, 56, and 59-74. We have jurisdiction under 35 U.S.C. § 6(b).

We reverse.

Claims 1 and 63 are illustrative.

1. A method for visually displaying a grouping of menu items within a graphical user interface, the method comprising:

rendering a toolbar user interface,

the toolbar user interface comprising one or more menu items, each menu item comprising one or more visual identifiers, at least one menu item comprising one or more submenu items comprising the one or more visual identifiers of the corresponding menu item to distinguish the submenu item from submenu items corresponding to other of the menu items;

rendering a submenu item using the one or more visual identifiers associated with the menu item to which the submenu item corresponds;

receiving a modification to one or more of the visual identifiers associated with a selected menu item or one of its corresponding submenu items;

detecting the modification; and

automatically triggering, based on the detected modification, a corresponding modification to the one or more visual identifiers associated with at least one other of the selected menu item and its corresponding submenu items.

63. A computer program, stored on a tangible computer readable medium, comprising:

a first code segment configured to produce a first submenu item with a color that is dependent upon a first color to be used in producing a corresponding first menu item, the first code segment being configured to programmatically trigger a change in the first submenu item color based on a detected change in the first color to be applied to the first menu item; and

a second code segment configured to produce a second submenu item with a color that is dependent upon a second color to be used in producing a corresponding second menu item and that differs from the first color, the second code segment being configured to programmatically trigger a change in the second submenu item color based on a detected change in the second color to be applied to the second menu item.

The Examiner relies on the following references as evidence of unpatentability.

Lenz US 6,029,196 Feb. 22, 2000

Apple, Inc. et al. (“Apple”), *Inside Macintosh: Macintosh Toolbox Essentials* (1992), Ch. 3, “Menu Manager,” <http://developer.apple.com/documentation/mac/Toolbox/Toolbox-2.html>

Claims 63-68 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Apple.

Claims 1-15, 17-19, 22-41, 43-45, 48-53, 56, 59-62, and 69-74 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Apple and Lenz.

Claims 16, 20, 21, 42, 46, 47, 54, 55, 57, and 58 have been canceled.

In the § 103(a) rejection of claim 63, the Examiner finds that Apple does not “specifically” teach any of the recitations of claim 63, less the preamble. (Ans. 23.)

However, this limitation [sic; the body of claim 63] would have been obvious to one of ordinary skill in the art at the time of the invention, [i]n view of Apple, because Apple teaches a process of changing menu colors and the ability to have any number of color combinations the users desire as well as any number of menu configurations as the user desires (Apple 3-50 - 3-52). Apple further teaches the ability to write an independent menu procedure and definition file in which the user can specify specific behaviors of the menu that can differ

from one menu to the next, which includes changing the color (Apple 3-87 and 3-88).

(Ans. 23.)

We can assume for the purposes of this appeal that the reference teaches all that the rejection attributes to it. The rejection does not, however, apply the teachings of Apple to the limitations of claim 63, nor offer any explanation of how or why the teachings that are deemed to reside in Apple might show that the claimed subject matter would have been obvious to one of ordinary skill in the art.

We have reviewed the reference, with particular emphasis on the pages identified in the rejection of claim 63, but we do not find disclosure or suggestion for all the limitations of the claim. We agree with Appellants that Apple at least does not disclose or suggest a code segment configured to programmatically trigger a change in a submenu item color based on a detected change in the color to be applied to the associated menu item.

Because the rejection fails to set forth a case for *prima facie* obviousness of claim 63, we do not sustain the rejection of claim 63, nor of claims 66 through 68, which depend from claim 63.

In the § 103(a) rejection of claim 1 over Apple and Lenz, the Examiner finds that Apple fails to “expressly” teach receiving a modification to one or more of the visual identifiers associated with a selected menu item or one of its corresponding menu items, detecting the modification, and automatically triggering, based on the detected modification, a corresponding modification to the one or more visual

identifiers associated with at least one other of the selected menu item and its corresponding submenu item. (Ans. 3.)

The rejection turns to Lenz for its teachings, in columns 4 and 5, relating to the use of a Java Archive (JAR) file, by which an administrator can automatically push out and update software on client systems (e.g., Fig. 6). The reference does not appear to disclose, contrary to the rejection, that “the jar file contains all user profile configurations for the client and in which preference settings that include menu options are automatically set (Lenz column 5, lines 47-60).” (Ans. 4.)

We can assume for purposes of this appeal that Lenz teaches all that the rejection of claim 1 attributes to the reference. We are not persuaded, however, that modifying the system of Apple to incorporate the “automatic software updates” as taught by Lenz (Ans. 4) would have resulted in the method of claim 1. In particular, Lenz has not been shown to have “changes reflect throughout the interface” (Ans. 4) in the manner required by claim 1.

Apple does teach menu items and submenu items (e.g., page 3-6 and pages 3-12 through 3-14). Apple does teach that the text and characteristics (such as color) of menu items can be modified by a person who designs the user interface (e.g., pages 3-97 through 3-100), and that available routines can modify entries in a menu color information table (page 3-102).

However, we agree with Appellants that neither Apple, nor Lenz, nor any combination thereof teaches the limitations that the Examiner finds not to be “expressly” taught by Apple in the rejection applied against claim 1.

The rejection of the remaining independent claims (26, 27, 31, 38, and 50) suffers similar deficiencies as the rejection applied against claim 1. We

Appeal 2008-0690
Application 10/112,029

thus do not sustain the rejection of claims 1-15, 17-19, 22-41, 43-45, 48-53, 56, 59-62, and 69-74 under 35 U.S.C. § 103(a) as being unpatentable over Apple and Lenz.

CONCLUSION

The Examiner's decision is reversed.

REVERSED

clj

FISH & RICHARDSON P.C.
P.O. BOX 1022
MINNEAPOLIS, MN 55440-1022