



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
14/284,304	05/21/2014	Robert Douglas Arnold	FACEP116	1016
107519	7590	09/30/2019	EXAMINER	
VAN PELT, YI & JAMES LLP AND FACEBOOK, INC. 10050 N. Foothill Blvd., Suite 200 Suite 200 Cupertino, CA 95014			BORJA, ROBERTO	
			ART UNIT	PAPER NUMBER
			2173	
			NOTIFICATION DATE	DELIVERY MODE
			09/30/2019	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

usptocorrespondence@ip-patent.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte ROBERT DOUGLAS ARNOLD,
JONATHAN M. KALDOR, and
DENIS KOROSKIN

Appeal 2019-000594
Application 14/284,304¹
Technology Center 2100

Before: ST. JOHN COURTENAY III, MARC S. HOFF, and
SCOTT B. HOWARD, *Administrative Patent Judges*.

HOFF, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF THE CASE

Appellant appeal under 35 U.S.C. § 134 from the Examiner's final rejection of claims 1-20.² We have jurisdiction under 35 U.S.C. § 6(b).

¹ We use the word "Appellant" to refer to "applicant" as defined in 37 C.F.R. § 1.42. Appellant identifies Avaya Inc. as that the real party in interest. App. Br. 2.

² Claims 3, 6–10, 13–15, and 18–20 stand rejected under 35 U.S.C. § 103(a) over Lewin, Fisher, Burbeck, and Santos but are not argued in the Appeal Brief. Claims 4 and 5 stand rejected under 35 U.S.C. § 103(a) over Lewin, Fisher, Burbeck, Santos, and Takashima but are not argued in the Appeal Brief. Appeal Br. 3.

We AFFIRM.

Appellant's invention is a technique for asynchronous execution of instructions for an application using a multi-threaded approach. A main thread handles execution of instructions to generate a hierarchy of layers representing a GUI, wherein each layer represents a logical grouping of components of the GUI. An input thread handles asynchronous execution of instructions to process user input based on interactions with the GUI. A graphics thread handles asynchronous execution of instructions to generate and/or update display output in relation to one or more layers of the GUI hierarchy. Spec. ¶ 4.

Claim 1 is exemplary of the claims on appeal:

1. A method comprising:

by a computing device, executing, by a main thread, instructions to generate a GUI hierarchy comprising a representation of a graphical user interface (GUI), the GUI hierarchy comprising a hierarchical organization of layers, wherein each layer represents a logical grouping of components of the GUI;

by the computing device, executing, by the main thread, instructions to provide copies of the GUI hierarchy to an input thread and a graphics thread, wherein the copies of the GUI hierarchy are respectively stored in memory reserved for the input thread and memory reserved for the graphics thread;

by the computing device, asynchronously executing, by the graphics thread, instructions to render, based on the copy of the GUI hierarchy stored in the memory reserved for the graphics thread, the GUI in relation to one or more layers of the GUI hierarchy;

by the computing device, asynchronously executing, by the input thread, instructions to process user input to determine a gesture based on data received from input devices, wherein the data indicates user interactions with at least one identified layer of the GUI hierarchy;

by the computing device, asynchronously executing, by the input thread, instructions to provide information about the user input to the main thread and to the graphics thread; and

by the computing device, asynchronously executing, by the graphics thread, instructions to update, based on the information about the user input received from the input thread, the GUI in relation to one or more layers of the GUI hierarchy.

The Examiner relies upon the following prior art in rejecting the claims on appeal:

Fisher et al. ("Fisher")	US 2009/0287824 A1	Nov. 19, 2009
Lewin et al. ("Lewis")	US 2013/0159893 A1	June 20, 2013
Santos	US 2014/0049502 A1	Feb. 20, 2014
Takashima	US 2014/0173429 A1	June 19, 2014

Burbeck, *Applications Programming in Smalltalk-80TM: How to use Model-Viewer-Controller (MVC)*, ParcPlace Systems, Inc., 1–20 (1992) (hereinafter "Burbeck").

Claims 1, 2, 11, 12, 16, and 17 stand rejected under 35 U.S.C. § 103 as being unpatentable over Lewin, Fisher, and Burbeck. Final Act. 5–32.

Claims 3, 6–10, 13–15, and 18–20 are rejected under 35 U.S.C. 103 as being unpatentable over Lewin, Fisher, Burbeck, and Santos. Final Act. 32–50.

Claims 4–5 are rejected under 35 U.S.C. 103 as being unpatentable over Lewin, Fisher, Burbeck, Santos and Takashima. Final Act. 50–53.

Throughout this decision, we make reference to the Appeal Brief filed April 4, 2018 (“App. Br.”); the Final Office Action mailed Feb. 7, 2017 (“Final Act.”); and the Examiner’s Answer mailed July 27, 2018 (“Ans.”) for their respective details.

ISSUE

Appellant’s arguments present us with the following issue:

1. Does the combination of Lewin, Fisher, and Burbeck teach or suggest execution of instructions by a main thread, and asynchronous execution of instructions by an input thread and a graphics thread?
2. Does the combination of Lewin, Fisher, and Burbeck teach or suggest executing, by the input thread, instructions to provide information about the user input to the main thread and to the graphics thread?

PRINCIPLES OF LAW

One cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. *In re Keller*, 642 F.2d 413, 425 (CCPA 1981). The test of obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference, nor is it that the claimed invention must be expressly suggested in any one or all of the references. *Id.* Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art. *Id.*

ANALYSIS

CLAIMS 1, 2, 11, 12, 16, AND 17

Appellant present arguments against the applied references (Lewin, Fisher, Burbeck) individually, which is generally ineffective to show error in a rejection based upon a combination of references. *See Keller*, 642 F.2d at 425. We nonetheless address Appellant’s arguments insofar as they allege the absence of a prima facie case of obviousness.

Appellant argues that Lewin does not disclose the use of distinct threads for user input computation, graphics rendering and UI hierarchy generation. App. Br. 7. Appellant’s arguments are not persuasive of Examiner error. The Examiner admits that Lewin does not teach separating computation of user input from graphics rendering. Final Act. 9. Rather, the Examiner cites Lewin for its teaching of a main thread and a combined input/graphics thread referred to as “UI rendering engine 902.” Lewin ¶¶ 80, 82; Ans. 4. We agree with the Examiner’s finding. Lewin teaches that UI rendering engine 902 “handles the logic and event handling associated with the UI elements.” *Id.* ¶ 82. It receives data from the UI client engine asynchronously, such that “transmission of data from the UI client engine 904 to the UI rendering engine 902 is independent of processing of data, or inputs, by the application.” Lewin ¶ 82 (emphasis omitted).

Appellant next argues that Fisher does not teach the use of a dedicated input thread to process user input information, in that Fisher’s I/O thread 230, cited by the Examiner, does not process any user input information. App. Br. 7–8. Appellant’s argument is not persuasive. First, Appellant does not support the allegation that I/O thread 230 does not process user input information with evidence. Second, we agree with the Examiner’s finding

that Fisher teaches a distinct main browser thread (main thread), an I/O thread (input thread) and a renderer thread (graphics thread). *Id.*; Fisher ¶ 35. Fisher teaches that I/O thread may be used by “browser” (main) process 220 to communicate with the rendering engine process, and may communicate messages from browser process 110 to one or a plurality of render views. Fisher ¶ 43. Fisher further teaches that “resource requests for content may be handled by I/O thread 230 and this may reduce the amount of resource handling needed to be done by the main browser thread 220.” Fisher ¶ 46 (emphasis omitted); Ans. 4.

Last, Appellant contends that Burbeck does not teach an input *thread*, but rather three types of *objects*. Appellant alleges that the Examiner conflates Burbeck’s controller object with an input thread, and that an object is a clearly distinct concept from a thread of execution. App. Br. 8.

This argument is not persuasive, because the Examiner does not rely on Burbeck for a teaching of threads. Ans. 5. Rather, the Examiner relies on Burbeck to suggest the claimed “executing, by the input thread, instructions to provide information about the user input to the main thread and the graphics thread.” Ans. 5. Burbeck’s controller module, analogized to the input thread, may receive input from a user and submit messages to the view module (graphics thread) and model module (main thread), in response to user input. “The controller interprets the mouse and keyboard inputs from the user, commanding the model and/or the view to change as appropriate.” Burbeck 4. The controller (input thread) may message the view module: “[t]he controller can assume responsibility for notifying the view of any changes because it interprets the user’s requests.” *Id.* “[T]he controller (input thread) may send particular commands to the model module

(main thread): ‘the model manages the behavior and data of the application domain, response to requests for information about its state (usually from the view), and responds to instructions to change state (usually from the controller).’” Ans. 5; *see also* Burbeck 4. We agree with the Examiner that the person having ordinary skill in the art would have been motivated to incorporate Burbeck’s model view controller architecture into the combination of Lewin and Fisher, to allow for more easily maintained code and provide a more responsive user interface. Final Act. 11.

We conclude that the Examiner did not err in combining Lewin, Fisher, and Burbeck to obtain the invention under appeal. We sustain the Examiner’s § 103(a) rejection.

CLAIMS 3-10, 13-15, AND 18-20

Appellant, in the brief, does not argue the rejection of claims 3-10, 13-15, and 18-20. 37 CFR 41.31(c) states in pertinent part, however, that “[a]n appeal, when taken, is presumed to be taken from the rejection of all claims under rejection unless cancelled by an amendment filed by the applicant and entered by the Office.”

No such amendment cancelling claims has been filed. Accordingly, we treat these claims as appealed by Appellant but not argued.

Therefore, we sustain the Examiner’s rejection of claims 3, 6-10, 13-15, and 18-20 under 35 U.S.C. § 103(a) as being unpatentable over Lewin, Fisher, Burbeck, and Santos.

We also sustain the Examiner’s rejection of claims 4 and 5 under 35 U.S.C. § 103(a) as being unpatentable over Lewin, Fisher, Burbeck, Santos, and Takashima.

CONCLUSION

1. The combination of Lewin, Fisher, and Burbeck suggests execution of instructions by a main thread, and asynchronous execution of instructions by an input thread and a graphics thread.

2. The combination of Lewin, Fisher, and Burbeck suggests executing, by the input thread, instructions to provide information about the user input to the main thread and to the graphics thread.

In summary:

Claims Rejected	Basis	Affirmed	Reversed
1-20	§ 103(a)	1-20	
Overall Outcome		1-20	

DECISION

The Examiner's decision to reject claims 1-20 is affirmed.

No time period for taking any subsequent action in connection with this Appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED