



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.
14/076,206	11/09/2013	Erik J. Burekart	RSW920130094US1 (797)	4267
46320	7590	02/15/2018	EXAMINER	
CRGO LAW STEVEN M. GREENBERG 7900 Glades Road SUITE 520 BOCA RATON, FL 33434			PAN, YONGJIA	
			ART UNIT	PAPER NUMBER
			2145	
			NOTIFICATION DATE	DELIVERY MODE
			02/15/2018	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):

docketing@crgolaw.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte ERIK J. BURCKART, MICHAEL P. ETGEN,
ANDREW J. IVORY, and DAVID M. STECHER

Appeal 2017-009014
Application 14/076,206¹
Technology Center 2100

Before BRUCE R. WINSOR, JEREMY J. CURCURI, and
PHILLIP A. BENNETT, *Administrative Patent Judges*.

BENNETT, *Administrative Patent Judge*.

DECISION ON APPEAL

Appellants appeal under 35 U.S.C. § 134(a) from the Examiner’s final rejection of claims 4–10.² We have jurisdiction under 35 U.S.C. § 6(b).

We affirm.

RELATED PROCEEDINGS

Appellants identify Appeal 2017-008594 (“the ’594 Appeal”), which is the subject appeal of Application 14/076,206 as a related appeal. App. Br. 2. The ’594 Appeal is decided concurrently with the present appeal.

¹ Appellants’ Brief (“App. Br.”) identifies International Business Machines Corporation as the real party in interest. App. Br. 2.

² Claims 1–3 were previously cancelled.

CLAIMED SUBJECT MATTER

The claims are directed to utilizing object cloning to enable nested drag and drop operations within a graphical user interface. Spec. Abstract. Claim 4, reproduced below, is illustrative of the claimed subject matter:

4. A graphical user interface (GUI) data processing system comprising:
 - a host computer with memory and at least one processor;
 - an operating system executing in the memory of the host computer;
 - an application hosted by the operating system and providing a GUI with different objects displayed thereon; and,
 - a nested object drag and drop module coupled to the GUI, the module comprising program code enabled upon execution in the memory of the host computer to respond to a drag event in the GUI by identifying an object amongst the objects in the GUI associated with the drag event, by retrieving into the memory one or more clones of other objects in the GUI that are nested in the identified object, by generating an avatar for the identified object and the one or more clones, and by moving a display of the avatar from a source location of the drag event to a target location of a drop event detected in the GUI of the application.

App. Br. 11 (Claims Appendix).

REJECTION

Claims 4–10 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Anthony (US 2007/0061745 A1, published Mar. 15, 2007) and “Draggable with Droppable Clone” as shown in JSFiddle (publicly available October 3, 2013 as evidence by Internet Wayback Machine) (“JSFiddle”). Final Act. 2–8.

ISSUES

Has the Examiner erred in finding Anthony and JSFiddle teach or suggest “retrieving in memory one or more clones of other objects in the GUI that are nested in the identified object,” as recited in independent claims 4 and 8?

ANALYSIS

In rejecting independent claims 4 and 8, the Examiner relies primarily on Anthony. The Examiner finds Anthony teaches the claimed invention except that “Anthony differs from the claims in that Anthony fails to teach the objects retrieved from memory are clones of objects.” Final Act. 4. The Examiner supplements the teachings of Anthony with those of JSFiddle, finding “retrieving clones of objects for a drag and drop operation is taught by JSFiddle.” *Id.* (citing JSFiddle source code). The Examiner concludes as follows:

Since Anthony and JSFiddle teach a system and method for drag and drop operations, it would have been obvious to one of skill[] in the art to modify Anthony to include JSFiddle’s clones to achieve the predictable result of replicating data for Anthony’s drag and drop.

Id.

Appellants contend “Anthony and JSFiddle clearly lack all claimed aspects of Appellants’ claims 4 and 8.” App. Br. 5. Appellants acknowledge that Anthony’s “Figure 4 shows nested folders,” but Anthony’s is deficient because “the folders are not clones of objects that are nested in an object that has been associated with a drag event.” App. Br. 6.

According to Appellants, the cited portions of Anthony are limited to teaching simultaneous display of container objects (Anthony ¶ 32) and dragging and dropping objects from various hierarchical levels in a GUI (Anthony ¶ 44). App. Br. 7. Appellants further argue the Examiner's additional reliance on JSFiddle does not account for the limitation "that are nested in the identified object." App. Br. 8. Finally, Appellants argue the action of dragging and dropping nested folders, as taught by Anthony, "does not account for the retrieval into memory of one or more clones of other objects in a GUI that are nested in an identified object." App. Br. 9.

We are not persuaded the Examiner has erred in concluding it would have been obvious to combine Anthony and JSFiddle in the manner recited in the independent claims. Essentially, Appellants' position is that the rejection is in error because Anthony does not teach "retrieving in memory one or more clones of other objects in the GUI that are nested in the identified object," and JSFiddle also does not teach "retrieving in memory one or more clones of other objects in the GUI that are nested in the identified object." This argument does not address the rejection made by the Examiner; it attacks the references singly where the rejection is based on their combined teachings.

The Examiner's rejection finds that Anthony teaches retrieving into memory nested objects associated with a drag and drop event. Ans. 7 (citing Anthony ¶¶ 32, 44). The Examiner relies on JSFiddle to show that it was known to retrieve clones of objects for a drag and drop operation. Taken together, the Examiner concludes because it was known to use cloning in drag and drop operations, it would have been obvious to add that functionality to Anthony's drag and drop events for nest objects. We find

the Examiner's findings and rationale well-supported by the evidence. Appellants argue the cited paragraphs in Anthony teach only the use of container objects and dragging and dropping from varying levels of hierarchy. We disagree that the teachings are so limited because "[i]t is well settled that a prior art reference is relevant for all that it teaches to those of ordinary skill in the art." *In re Fritch*, 972 F.2d 1260, 1264 (Fed. Cir. 1992).

Here, given Anthony's disclosure of nested objects and drag/drop operations provided in paragraphs 32 and 44, we agree with the Examiner that a person of ordinary skill in the art would have appreciated that Anthony's nested objects could be dragged/dropped within a GUI, and would have further appreciated that in carrying out that process, the nested objects would be retrieved into memory. Accordingly, we are not persuaded the Examiner failed to map the claim language "that are nested in the identified object" to the prior art, as Appellants have alleged. As a result, we conclude the Examiner has not erred in rejecting independent claims 4 and 8 under 35 U.S.C. § 103(a), and we sustain the rejection of these claims. Appellants do not present separate arguments for patentability of dependent claims 5–7 and 9–10. As a result, the dependent claims fall with their respective independent claims.

DECISION

We affirm the Examiner's rejection of claims 4–10.

Appeal 2017-009014
Application 14/076,206

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

AFFIRMED