



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.
13/335,648	12/22/2011	William Brandon George	058083-0857843(B1438US01)	7886
72058	7590	10/05/2020	EXAMINER	
Kilpatrick Townsend & Stockton LLP/Adobe Adobe Systems, Inc. 58083 Mailstop: IP Docketing - 22 1100 Peachtree Street, Suite 2800 Atlanta, GA 30309-4530			YUN, CARINA	
			ART UNIT	PAPER NUMBER
			2194	
			NOTIFICATION DATE	DELIVERY MODE
			10/05/2020	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):

KTSDocketing2@kilpatrick.foundationip.com
ipefiling@kilpatricktownsend.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte WILLIAM BRANDON GEORGE and KEVIN G. SMITH

Appeal 2019-003362
Application 13/335,648
Technology Center 2100

Before JOSEPH L. DIXON, SCOTT B. HOWARD, and
STEVEN M. AMUNDSON, *Administrative Patent Judges*.

HOWARD, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF THE CASE

Pursuant to 35 U.S.C. § 134(a), Appellant¹ appeals from the Examiner's decision to reject claims 1, 2, 4–7, 9–18, 20, 21, 23, 24, and 26–39. *See* Final Act. 1. Claims 3, 8, 19, 22, and 25 have been cancelled. *See* Appeal Br. 21–26.² We have jurisdiction under 35 U.S.C. § 6(b).

We AFFIRM IN PART.

¹ We use the word Appellant to refer to “applicant” as defined in 37 C.F.R. § 1.42. Appellant identifies the real party in interest as Adobe Systems, Inc. Appeal Br. 2.

² We note the Appeal Brief is not paginated. We number the pages starting with page 1 on the first page after the Table of Contents.

CLAIMED SUBJECT MATTER

The claims are directed to systems and methods relating to unsupported user interface actions. Claim 1, reproduced below, is illustrative of the claimed subject matter:

1. A method, comprising:

receiving, by an application hosted on a computing device, a user interface event at a user interface of the application, wherein the application uses an event handling hierarchy of an unsupported event module and event handlers, wherein the unsupported event module is installed in the application as a plug-in;

determining, by the unsupported event module, whether any of the event handlers is capable of supporting the user interface event;

in response to determining that the event handlers are incapable of handling the user interface event, identifying, by the unsupported event module, that the user interface event is an unsupported event, wherein operations of the user interface of the application are unaffected by the unsupported event;

generating, by the unsupported event module, one or more records comprising information about the user interface event and an indication that the user interface event is unsupported; and

storing, by the unsupported event module, the one or more records at the computing device;

transmitting the one or more records from the computing device to a collection server;

modifying the application on the computing device based on an analysis of the one or more records and of a plurality of records collected at the collection server and associated with instances of the application on a plurality of computing devices, the modifying comprising an update to the event handling hierarchy to include an event handler configured to handle the user interface event and support an additional operation of the application corresponding to the user interface event.

REFERENCES

The prior art relied upon by the Examiner is:

Name	Reference	Date
Wu	US 6,741,967 B1	May 25, 2004
Larsson	US 2004/0107387 A1	June 3, 2004
Golde	US 6,951,022 B1	Sept. 27, 2005
Yavilevich	US 7,941,525 B1	May 10, 2011

Brereton, R. G. (1994). Tutorial review. Object oriented programming on personal computers. *The Analyst*, 119(10), 2149-2160.
doi:10.1039/an9941902149 (Brereton)

REJECTIONS

Claims Rejected	35 U.S.C. §	Reference(s)/Basis
37, 38	112	Written Description
1, 2, 4, 5, 7, 9, 10, 15–17, 20, 21, 23, 33–39	103(a)	Brereton, Larsson Golde, Wu,
6, 11–14, 18	103(a)	Brereton, Larsson, Golde, Wu, ClickTale,
24, 26–32	103(a)	Yavilevich, Brereton, Golde, Wu, Larsson

OPINION

We have reviewed the Examiner's rejection in light of Appellant's arguments that the Examiner erred. In reaching our decision, we have considered all evidence presented and all arguments made by Appellant. With the exception of the written-description rejection of claim 37, we are persuaded by Appellant's arguments that the Examiner erred.

Written-Description Rejections

Claim 37

The Examiner finds that the Specification does not provide written-description support for the phrase “an attribute updatable by the plurality of event handlers” as recited in claim 37. Final Act. 3; *see also id.* at 22; Ans. 27. Specifically, the Examiner finds that “although the specification describes an attribute associate with the user interface event may indicate that the event has been handled, it does not state that these attributes are updated by the plurality of event handlers.” Ans. 27 (emphasis in original). The Examiner further finds that although “[t]he attributes may be checked once . . . this does not mean they are updated by the event handlers.” *Id.* Rather, the Examiner finds that “[t]here is no way to determine if the attributes are updated by the event handlers or if it is the same attribute being checked.” *Id.*

Appellant argues paragraph 35 of the Specification “describes in part that ‘an attribute may be associated with the user interface event to indicate that the user interface event has been handled. In such an embodiment, an unsupported event handler may check this attribute to determine whether or not the application handled the event.’” Appeal Br. 7 (quoting Spec. ¶ 35). Appellant further argues that a person having ordinary skill in the art

would have easily understood that associating an attribute with a user interface event to indicate that the user interface event and subsequently checking this attribute to determine whether or not the application handled the event means that an attribute indicating that a certain user interface event has not been handled may later be updated to indicate that the user interface event has been handled, when applicable.

Id. (emphasis in original) Appellant also argues that the claims do not require the attributes to be updated, just that it is “updatable.” Reply Br. 2.

To satisfy the written-description requirement, the disclosure must reasonably convey to skilled artisans that Appellant possessed the claimed invention as of the filing date. *See Ariad Pharms., Inc. v. Eli Lilly & Co.*, 598 F.3d 1336, 1351 (Fed. Cir. 2010) (en banc). Specifically, the description must “clearly allow persons of ordinary skill in the art to recognize that [the inventor] invented what is claimed.” *Id.* (alteration in original) (quoting *Vas-Cath Inc. v. Mahurkar*, 935 F.2d 1555, 1563 (Fed. Cir. 1991)). Thus, an applicant complies with the written-description requirement “by describing the invention, with all its claimed limitations, not that which makes it obvious,” and by using “such descriptive means as words, structures, figures, diagrams, formulas, etc., that set forth the claimed invention.” *Lockwood v. Am. Airlines, Inc.*, 107 F.3d 1565, 1572, (Fed. Cir. 1997). The claimed invention need not be recited *in haec verba* in the original specification in order to satisfy the written-description requirement. *Ariad*, 598 F.3d at 1352.

We are not persuaded by Appellant’s argument that the Examiner erred. Paragraph 35 of the Specification states that

In such an embodiment, locating one or more unsupported event handlers at the end of the application’s event handling hierarchy is sufficient to indicate that the event was not otherwise handled. In other embodiments where events may be further propagated after being handled in the application, *an attribute may be associated with the user interface event to indicate that the user interface event has been handled.* In such an embodiment, an unsupported event handler may check this attribute to determine whether or not the application handled the event.

Spec. ¶ 35 (emphasis added). Although the Specification describes an attribute indicating whether an event had been handled, the Specification is silent as to how that attribute is associated with the user interface event. Although it might have been obvious for the event handler to update the attribute, that is not sufficient for written-description support. *See Lockwood*, 107 F.3d at 1572.

Nor do we give any weight to Appellant's assertion regarding how a person having ordinary skill in the art would have understood that paragraph. Appellant cites no evidentiary support for that argument. *See* Appeal Br. 7. It is well settled that mere attorney arguments and conclusory statements, which are unsupported by factual evidence, are entitled to little probative value. *See Johnston v. IVAC Corp.*, 885 F.2d 1574, 1581 (Fed. Cir. 1989) ("Attorney's argument is no substitute for evidence."); *see also In re Pearson*, 494 F.2d 1399, 1405 (CCPA 1974) (attorney argument is not evidence).

Accordingly, we sustain the Examiner's rejection of claim 37 for failure to comply with the written-description requirement.

Claim 38

The Examiner finds that the Specification does not provide written-description support for "the user interface event is received at a location on the user interface corresponding to a user interface object" and "that the plurality of event handlers are incapable of supporting the user interface event" as recited in claim 38. Final Act. 3, 22; Ans. 27–28. More specifically, the Examiner finds that "[i]ncapable and not handling an event for recording are different." Final Act. 22. According to the Examiner, the

Specification only discloses “events that do not have a corresponding event handler” but that “does not mean that those handlers are incapable of supporting the event if they were programmed to handle the event.” Ans. 27–28.

With regard to the location of user interface event, Appellant directs us to paragraph 36 of the Specification. Appeal Br. 38; Reply Br. 3. With regard to incapable, Appellant directs us to paragraph 22 of the Specification. Appeal Br. 8; Reply Br. 3. According to Appellant, “if an application includes event handlers designed to support various user interface events, but none of those event handlers corresponds to a particular type of user interface event” as described in paragraph 22, “it follows that the included event handlers are not capable of supporting that particular type of event.” Appeal Br. 8. Appellant also argues that “the mere possibility of programming an object in a different way does not mean that the object is capable of performing a particular action without such programming being present.” Reply Br. 3.

We are persuaded by Appellant’s arguments that the Examiner erred. Paragraph 36 of the Specification describes clicking a button on interface. The button is at a specific location on the interface and the click is a user interface event that takes place at that location. Accordingly, the Specification provides written-description support for “the user interface event is received at a location on the user interface corresponding to a user interface object” as recited in claim 38.

Additionally, paragraph 22 of the Specification provides sufficient support for “user interface events” that are not supported by the application. The Examiner has not sufficiently explained how an event which is not

supported by the application is different from having a “plurality of event handlers are [that] incapable of supporting the user interface event” as recited in claim 38. Although the Examiner hypothesizes that in an alternate version the application described in the Specification could be modified such that it no longer provided support for the claim limitation, we focus on our analysis on the Specification as written.

Additionally, it is inapposite that the Specification does not use the word “incapable.” The claimed invention need not be recited *in haec verba* in the original specification in order to satisfy the written-description requirement. *Ariad*, 598 F.3d at 1353.

Accordingly, based on the current record, we reverse the Examiner’s rejection of claim 38 for failing to comply with the written description requirement.

Obviousness Rejections

Claim 1 recites “determining, by the unsupported event module, whether any of the event handlers is capable of supporting the user interface event” and “in response to determining that the event handlers are incapable of handling the user interface event, identifying, by the unsupported event module, that the user interface event is an unsupported event, wherein operations of the user interface of the application are unaffected by the unsupported event.” Appeal Br. 21. The Examiner finds Brereton teaches those limitations. Final Act. 4–5 (citing Brereton 2154). Alternatively, the Examiner finds Brereton teaches an “unsupported event” and Larsson teaches “the ability to store and capture events.” Ans. 28–29 (citing Brereton 2154; Larsson ¶ 9). According to the Examiner, “it is the

combination of references which yield the claimed invention, as Larsson captures all the exception, including unhandled ones, which is consistent with applicant's specification." *Id.* at 29.

Appellant argues "Brereton generally explains the concept of an event, but makes no mention of an '*unsupported event module*.'" Appeal Br. 10 (emphasis in original) (discussing Brereton 2153). Appellant further argues that "Brereton explains that an event is passed along through a hierarchy of event handlers until the appropriate event handler recognizes and services the event. *If an event is not recognized by any event handler, it is simply ignored.*" *Id.* (emphasis in original) (discussing Brereton 2154).

Appellant further argues that Larsson does not cure Brereton's deficiency. *See* Appeal Br. 11–13. More specifically, Appellant argues

Larsson describes a database that stores "unhandled exceptions." These exceptions relate to exceptions and bugs in the code that result in the crash of an application. In other words, the systems and methods disclosed by Larsson cannot record and report these "unhandled exceptions" unless a crash of the application occurs, *i.e.*, the application stops working and its user interface becomes inoperable. For example, Larsson at [0012]-[0013] describes how the information recorded by the system is used to generate error reports and that "[t]he error reports may assist the developer in correcting errors occurring within the client computer 2."

Id. at 11–12 (footnotes omitted). According to Appellant, "[t]he system of Larsson is not designed to and is not capable of detecting and taking action upon unsupported user interface events that occur with no change to the operations of the software application." *Id.* at 12.

We are persuaded by Appellant's arguments that the Examiner erred in finding Brereton alone or Brereton in combination with Larsson teaches

the recited “unsupported event module.” Specifically, we agree with Appellant that Brereton simply teaches ignoring unsupported user interface events and does not teach or suggest an unsupported event module that determines “whether any of the event handlers is capable of supporting the user interface event” and identifies unsupported events as recited in claim 1. *See* Brereton 2154 (“Some objects do not have associated event handlers. . . . *Note that if there is no event handler for a given action at higher levels of the hierarchy, the event is ignored.*” (emphasis added)).

We similarly agree with Appellant that Larsson does not cure that deficiency. Although the Examiner finds that “Larsson teaches the ability to store and capture events, such as the unsupported events” (Ans. 29 (citing Larsson ¶ 9)), the Examiner does not sufficiently explain how Larsson, either alone or in combination with Brereton, teaches all of the requirements of the unsupported event module recited in claim 1. Specifically, the cited portion of Larsson is silent as to how an unsupported event is determined. That is, the cited portion of Larsson is silent as to whether an unsupported event is determined because it is ignored (like in Brereton) or that an unsupported event modules “determin[es] . . . whether any of the event handlers is capable of supporting the user interface event” as recited in claim 1. Accordingly, the Examiner has not supplied a sufficient factual basis to support the rejection. *See In re Caveney*, 761 F.2d 671, 674 (Fed. Cir. 1985) (Examiner’s burden of proving non-patentability is by a preponderance of the evidence); *see also In re Warner*, 379 F.2d 1011, 1017 (CCPA 1967) (“The Patent Office has the initial duty of supplying the factual basis for its rejection. It may not, because it may doubt that the

invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in its factual basis.”).

Therefore, we are constrained on this record to reverse the Examiner’s rejection of claim 1, along with the rejections of claims 10 and 16, which recite limitations commensurate in scope to the disputed limitations discussed above, and dependent claims 2, 4, 5, 7, 9, 15, 17, 20, 21, 23, and 33–39.

Moreover, because the Examiner has not shown that Yavilevich or ClickTale cures the foregoing deficiencies regarding the rejection of the independent claims, we will not sustain the obviousness rejection of dependent claims 6, 11–14, 18, 24, and 26–32 for similar reasons.

CONCLUSION

The Examiner’s rejection of claim 37 for failing to comply with the written description requirement is affirmed.

The Examiner’s rejection of claim 38 for failing to comply with the written description requirement is reversed.

The Examiner’s rejections of claims 1, 2, 4–7, 9–18, 20, 21, 23, 24, and 26–39 as unpatentable under 35 U.S.C. § 103(a) are reversed.

DECISION SUMMARY

Claims Rejected	35 U.S.C. §	Reference(s)/Basis	Affirmed	Reversed
37, 38	112	Written Description	37	38
1, 2, 4, 5, 7, 9, 10, 15–17, 20, 21, 23, 33–39	103(a)	Brereton, Larsson, Golde, Wu		1, 2, 4, 5, 7, 9, 10, 15–17, 20, 21, 23, 33–39

Claims Rejected	35 U.S.C. §	Reference(s)/Basis	Affirmed	Reversed
6, 11–14, 18	103(a)	Brereton, Larsson, Golde, Wu, ClickTale		6, 11–14, 18
24, 26–32	103(a)	Yavilevich, Brereton, Golde, Wu, Larsson		24, 26–32
Overall Outcome			37	1, 2, 4–7, 9–18, 20, 21, 23, 24, 26–36, 38, 39

TIME PERIOD FOR RESPONSE

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 IN PART