

NOTE: This disposition is nonprecedential.

**United States Court of Appeals
for the Federal Circuit**

MICROSOFT CORPORATION,
Appellant

v.

UNILOC 2017 LLC,
Appellee

2021-2039

Appeal from the United States Patent and Trademark Office, Patent Trial and Appeal Board in No. IPR2020-00023.

Decided: October 20, 2022

ANDREW M. MASON, Klarquist Sparkman, LLP, Portland, OR, argued for appellant. Also represented by SARAH ELISABETH JELSEMA, DERRICK WADE TODDY.

NATHAN K. CUMMINGS, Etheridge Law Group, Southlake, TX, argued for appellee. Also represented by JAMES ETHERIDGE, BRIAN MATTHEW KOIDE, RYAN S. LOVELESS, BRETT MANGRUM.

Before LOURIE, DYK, and HUGHES, *Circuit Judges*.

HUGHES, *Circuit Judge*.

Microsoft appeals a decision of the Patent Trial and Appeal Board determining that Microsoft did not prove by a preponderance of evidence that claims 1–4, 6–14, 16–21 of U.S. patent No. 6,467,088 are unpatentable. Because substantial evidence does not support the Board’s factual findings, we vacate and remand.

I

Uniloc owns the ’088 patent, which is directed to techniques for upgrading or reconfiguring software and hardware components of electronic devices. Before updating components of electronic devices, it is generally necessary to assess compatibility with the rest of the device to determine whether the new component will cause problems. The ’088 patent solves potential compatibility conflicts by comparing “the needed and currently implemented components with previously stored lists of known acceptable and unacceptable configurations for the electronic device.” ’088 patent at 2:38–41. Claim 1 is representative:

1. A processor-implemented method for controlling the reconfiguration of an electronic device, the method comprising the steps of:

receiving information representative of a reconfiguration request relating to the electronic device;

determining at least one device component required to implement the reconfiguration request;

comparing the determined component and information specifying at least one additional component currently implemented in the electronic device with at least one of a list of known acceptable configurations for the electronic device and a list of known

unacceptable configurations for the electronic device; and

generating information indicative of an approval or a denial of the reconfiguration request based at least in part on the result of the comparing step.

Id. at 6:43–59.

Microsoft petitioned for inter partes review of claims 1–4, 6–14, and 16–21 of the '088 patent based on obviousness grounds. The Board instituted review but disagreed with all asserted grounds and concluded that Microsoft failed to show by a preponderance of evidence that claims 1–4, 6–14, 16–21 of the '088 patent would have been unpatentable as obvious.

II

We review the ultimate conclusion of obviousness de novo and “the Board’s factual findings underlying those determinations for substantial evidence.” *In re Ethicon, Inc.*, 844 F.3d 1344, 1349 (Fed. Cir. 2017).

The dispositive issue here was whether a prior art reference, Apfel (U.S. Patent No. 5,974,454), performs the claimed “comparing” step, which involves (1) a comparison between “a determined component” (i.e., the component required to implement the reconfiguration request), (2) “information specifying at least one additional component currently implemented in the electronic device,” and (3) “a list of known acceptable configurations.” '088 patent at 6:51–56. The Board found that Apfel did not disclose the comparing step. That conclusion lacks substantial evidence.

A

The Board erred in its factual finding regarding Apfel because it overlooked a passage that specifically discloses assessing the compatibility of available upgrades:

The servers are responsible for assessing whether an upgrade is available and whether it should be downloaded based on the information sent by computer **20**. For example, even if an upgrade is available, it *should not be downloaded* if the computer **20** already has the upgrade or if the upgrade is *somehow incompatible* with computer **20**.

'454 patent at 7:13–19 (italicizations added).

The Board cited the above passage in a parenthetical and noted that “Apfel would not allow the download of a version [] that is *incompatible* with computer 20[.]” J.A. 22 (emphasis added). But despite this citation, the Board failed to explain why this passage from Apfel did not disclose the required compatibility check. Indeed, the Board’s description of this passage contradicts its conclusion that “Apfel’s database lookup only determines that a new upgrade is available—not that there is a known *compatible* upgrade available.” J.A. 16–17 (internal quotation omitted).

In addition, the Board’s conclusion that Apfel does not teach a comparing step appears to be contradicted by other Apfel passages that it considered. The following passage, although not explicitly referring to a compatibility check, recognizes that different update packages correspond, for example, to different operating systems, and that a database of the different configurations is maintained to guide downloads:

At decision step **427**, it is determined whether there is an upgrade package for the Web Authoring Components program module. In the exemplary embodiment, the database server **80a** uses the information received in the HTTP query at step **415** to determine if an upgrade package is available, such as by a database lookup. Different update packages may be provided for different version combinations, different operating systems, and different languages. Thus, the database server **80a**

maintains a database of upgrade packages and corresponding configurations which should result in their download.

'454 patent at 9:30–40.

The Board interpreted this passage to mean that “the database server of Apfel maintains upgrade packages and corresponding configurations that *should* be downloaded.” J.A. 20 (emphasis added). It also determined that the “should result in their result download” language is used to reflect a user’s choice of whether to accept the invitation to download the package. J.A. 25–26. Even if the “should” language leaves room for ambiguity, the Board’s interpretation differs from the preceding sentence—“[d]ifferent update packages may be provided for different version combinations, different operating systems, and different languages.” ’454 patent at 9:36–38. That portion of Apfel, at a minimum, suggests a form of compatibility assessment to find the correct upgrade package and, combined with the other passage cited further above that specifically references incompatibility, renders the Board’s conclusion that Apfel does not disclose a compatibility check lacking in substantial evidence.

B

The Board also erred when it concluded that Apfel did not disclose the comparing step because it did not perform the determining and comparing steps in a certain order required by the disputed claims. *See* J.A. 18–19 (explaining that while Apfel compares the query and lookup table, “it is after the database lookup that a ‘determined component’ may be obtained”); J.A. 21 (explaining that Apfel had not “performed a compatibility determination *in the manner claimed*” (emphasis added)). In other words, according to the Board, Apfel does not perform a compatibility check after determining the availability of an upgrade.

We agree with Microsoft that the Board misconstrued the claims to require that the comparing and determining

step are performed by two separate acts in a certain order. Appellant's Br. 46. Nothing in the intrinsic record requires such a narrow construction, and neither party presented this construction to the Board.¹ Without any such evidence, we decline to impose such a narrow claim construction.

We review claim construction based on intrinsic evidence de novo. *Teva Pharm. USA, Inc. v. Sandoz, Inc.*, 574 U.S. 318, 331 (2015). A claim requires an ordering, when steps of a method actually recite an order, or when claim language, as a matter of logic, requires that the steps be performed in the order written. *Interactive Gift Express, Inc. v. Compuserve Inc.*, 256 F.3d 1323, 1342 (Fed. Cir. 2001); *Mformation Techs., Inc. v. Rsch. in Motion Ltd.*, 764 F.3d 1392, 1398 (Fed. Cir. 2014).

Here, the claim does not explicitly recite an order for performing the claim steps. Instead, Uniloc argues that the antecedent of the determined component is the determining step, because as a matter of logic, the component (i.e., newer version) must be determined before it can be compared. Uniloc's logic is flawed. Following Uniloc's logic to its conclusion, the receiving step would need to occur prior to the determining step, because the determining step requires information from the reconfiguration request and is listed prior to the receiving step in the representative claim. But the Board's construction requires that receiving and determining steps to occur in a reverse order. When the reconfiguration manager receives an upgrade request, the request must already include the user's preferred version, the determined component. '088 patent at 4:12–15; *see* J.A. 4.

¹ Uniloc claims that "Microsoft was on notice of Uniloc's position that the 'comparing' step had to be done after and separately from the 'determining' step." Appellee's Br. 15 n.3. But Uniloc included no supporting references for this claim, and the Board's decision did not discuss Uniloc's position.

Apfel was deemed not to disclose a comparing step specifically because the reconfiguration request in Apfel does not include the determined component. Hence, the claim cannot require that the steps be performed in the order written, but rather allows for the determining and comparing steps be part of a single process.

Moreover, the '088 patent provides examples where a reconfiguration manager performs the determining step *after* the comparing step. If, at the comparing step, the reconfiguration manager finds that the determined component corresponds to one of the known bad configurations, the reconfiguration manager repeats the determining step to find a set from known good configurations. '088 patent at 4:64–53. The reconfiguration manager can also receive “requests for an upgrade to a particular device feature,” which will require the manager to determine “several device components” to be upgraded. *Id.* at 4:56–61. Any construction that would narrow the determining and comparing steps to a certain order is not supported by either the claim language or the specification. The Board’s apparent construction to the contrary is reversed.

Because the Board erred in concluding that Apfel does not perform the claimed “comparing” step and implicitly relied on an improper claim construction, we vacate the Board’s decision and remand for further proceedings consistent with this opinion.

VACATED AND REMANDED

COSTS

Costs to Microsoft Corporation.