NOTE: This disposition is nonprecedential.

United States Court of Appeals for the Federal Circuit

IRON OAK TECHNOLOGIES, LLC,
Appellant

v.

MICROSOFT CORPORATION, SAMSUNG ELECTRONICS CO., LTD., GOOGLE LLC,
Appellees

2020-1701, 2020-1706, 2020-1707

Appeals from the United States Patent and Trademark Office, Patent Trial and Appeal Board in Nos. IPR2018-01552, IPR2018-01553, IPR2019-00106.

Decided: May 24, 2021

ALBERT BERTON DEAVER, JR., McAughan Deaver PLLC, Houston, TX, for appellant. Also represented by ROBERT JAMES MCAUGHAN, JR.

RICHARD ALAN CEDEROTH, Sidley Austin LLP, Chicago, IL, for appellee Microsoft Corporation. Also represented by SCOTT BORDER, JOSEPH A. MICALLEF, Washington, DC.

JOSEPH PALYS, Paul Hastings LLP, Washington, DC,

for appellee Samsung Electronics Co., Ltd. Also represented by PHILLIP W. CITROEN, STEPHEN BLAKE KINNAIRD, NAVEEN MODI, ANDERSON TO.

MATTHEW A. SMITH, Smith Baluch LLP, Menlo Park, CA, for appellee Google LLC. Also represented by ANDREW BALUCH, Washington, DC.

Before LOURIE, MAYER, and O'MALLEY, *Circuit Judges*.

O'MALLEY, *Circuit Judge*.

Iron Oak Technologies, LLC appeals from three final written decisions of the Patent Trial and Appeal Board ("Board"), each holding claim 1 of U.S. Patent No. 5,699,275 ("the '275 patent") unpatentable over the prior art. *Microsoft Corp. v. Iron Oak Techs., LLC*, No. IPR2019-00106, 2020 WL 633707 (P.T.A.B. Feb. 7, 2020); *Samsung Elecs. Co. v. Iron Oak Techs., LLC* (*Samsung I*), No. IPR2018-01552, 2020 WL 633816 (P.T.A.B. Feb. 7, 2020); *Samsung Elecs. Co. v. Iron Oak Techs., LLC* (*Samsung II*), No. IPR2018-01553, 2020 WL 633822 (P.T.A.B. Feb. 7, 2020). For the reasons explained below, we *affirm* as to all three decisions.

I. BACKGROUND

The '275 patent is entitled "System and Method for Remote Patching of Operating Code Located in a Mobile Unit." '275 patent, at [54]. According to its specification, software suppliers and other sellers of computer systems often need to correct or upgrade the existing software that their customers use. *Id.* at col. 1, ll. 13–15. But common methods of doing so in the prior art had several disadvantages. *Id.* at col. 1, ll. 15–23. Distributing floppy disks, for instance, was time-consuming and forced customers to use old software while waiting for updates. *Id.* at col. 1, ll. 17–20. Providing modem support to manually upgrade software was also time-consuming, as well as expensive

and prone to human error. *Id.* at col. 1, ll. 20–23. And a third prior art solution described in the specification provided patches only to remote systems at fixed locations over a single, continuous, interactive, and bidirectional communication link. *See id.* at col. 1, ll. 29–42.

To substantially reduce or eliminate the disadvantages of prior art systems and methods for updating software, the '275 patent provides a system for remote patching of operating code located in a mobile unit. *Id.* at col. 1, ll. 46–53. One embodiment of the system includes a communication network, a manager host, and several mobile units. *Id.* at col. 2, ll. 60–65; col. 3, ll. 10–11, 17–23. To enhance or correct a mobile unit's current operating code, the manager host can transmit a patch file that defines one or more patches in a set of discrete patch messages that are suitably sized for transmission through the communication network. *Id.* at col. 3, ll. 42–44, 57–62. On receipt of the patch messages, a mobile unit can verify them, merge the defined patches with its current operating code, and switch execution to the patched operating code. *Id.* at col. 3, ll. 63–66.

According to the '275 patent, the manager host can address patch messages to mobile units as appropriate for the patch file, including "to one of the mobile units, to all of the mobile units, or to a group of mobile units." *Id.* at col. 5, ll. 15–19; *see id.* at col. 3, ll. 61–62; col. 4, ll. 12–20. For instance, one embodiment groups five mobile units into a pair and a trio, and the specification explains that the manager host can address a patch message to only the group of two mobile units. *See id.* at col. 3, ll. 17–23; col. 5, ll. 20–25. By doing so, the pair of mobile units can have a different version of operating code than the trio of mobile units. *Id.* at col. 4, ll. 11–20.

The '275 patent also provides a method for remote patching of operating code located in a mobile unit. *Id.* at col. 1, ll. 62–63. According to one embodiment of the method, as depicted by the flowchart of Figure 5, a mobile

unit receives an initial patch message that includes a software version. *Id.* at col. 10, ll. 22–24, 33–34. The mobile unit then compares the software version of the initial patch message to the software version of its current operating code. *Id.* at col. 10, ll. 36–40. If the operating code's version is appropriate for the patch, the mobile unit proceeds to, *inter alia*, check the validity of the message, create patched operating code, and switch execution to the patched operating code. *See id.* at col. 10, l. 46–col. 11, l. 42. But, if the operating code's version is not appropriate for the patch, the mobile unit transmits an error message, *e.g.*, to the manager host. *Id.* at col. 10, ll. 42–45.

The '275 patent contains numerous system and method claims. Only independent claim 1 is at issue on appeal:

> A system for remote patching of operating code located in a mobile unit, comprising:
>
> a manager host operable to initiate transmission through a wireless communication network of at least one discrete patch message defining at least one patch;
>
> a first mobile unit operable to receive the at least one discrete patch message, the first mobile unit further operable to create patched operating code by merging the at least one patch with current operating code located in the first mobile unit and to switch execution to the patched operating code; and
>
> a second mobile unit operable to receive the at least one discrete patch message, the second mobile unit further operable to create patched operating code by merging the at least one patch with current operating code located in the second mobile unit and to switch execution to the patched operating code; and
>
> wherein the manager host is further operable to address the at least one discrete patch message such

> that the at least one discrete patch message is transmitted to the first mobile unit but not to the second mobile unit.

Id. at col. 13, ll. 32–53. Although not directly at issue on appeal, dependent claim 14 recites: "[t]he system of claim 1, wherein the manager host is further operable to address the at least one discrete patch message such that the at least one discrete patch message is transmitted to the first mobile unit and to the second mobile unit." *Id.* at col. 14, ll. 22–26.

Only one prior art reference is necessary to resolve these appeals: Japanese Patent Application No. 05-128022 ("Sugita").[1] Sugita describes a method of updating software on multiple mobile communications terminals using wireless communication. *See* J.A. 390 (¶ 12). Sugita's method has an initial stage and a final stage. *See id.* In the initial stage, updates of mobile communications terminals occur "in group units." *Id.* Specifically, Sugita describes a base station that transmits update information to each mobile communications terminal "based on the group ID of the group unit (all units, or the unit addresses belonging to a specific group)." *Id.* (¶ 13); *see* J.A. 391 (¶ 26) (describing a one-to-many communication format using group IDs for updating terminals "in group units"); J.A. 393 (¶ 49). In the final stage, updates occur "individually" and by individual ID, "one at a time." J.A. 390 (¶¶ 12–13). In one embodiment with multiple mobile communications terminals, each terminal—m1, m2, m3, m4 . . . —"targeted for

---

[1] Consistent with MPEP § 901.05(a), both Microsoft's and Samsung's translations convert Sugita's application number to No. 1993-128022. We note that the two translations are not entirely identical. But Iron Oak does not identify any substantive differences between the two translations. We refer to and quote from Microsoft's version.

update[s]" is "listed up" prior to the initial stage of transmitting update information by group ID. J.A. 392 (¶¶ 34–35).

Microsoft petitioned for *inter partes* review of claim 1 of the '275 patent, and Samsung and Google each filed two additional petitions challenging that claim. The Board instituted review of Microsoft's petition and Samsung's two petitions and granted Google's motions to join each of Samsung's petitions. In instituting review of Microsoft's petition, the Board construed "wherein the manager host is further operable to address the at least one discrete patch message such that the at least one discrete patch message is transmitted to the first mobile unit but not to the second mobile unit"—the "wherein" limitation—to require the manager host be "further operable to decide which specific mobile unit to send the at least one discrete patch message to before beginning transmission" of the message. *Microsoft*, 2020 WL 633707, at *3. The Board adopted a substantially similar construction of the limitation in *Samsung I* and *Samsung II*. *Samsung I*, 2020 WL 633816, at *4; *Samsung II*, 2020 WL 633822, at *3.

The Board's final written decisions found claim 1 of the '275 patent unpatentable over the prior art. In *Microsoft*, the Board found that Sugita anticipates claim 1 of the '275 patent. 2020 WL 633707, at *9. Two of its determinations are pertinent to this appeal. First, the Board maintained its construction of the wherein limitation. *Id.* at *3. It found no evidence supporting Iron Oak's arguments that (1) at the time of transmission of the patch message to the first mobile unit, the second mobile unit must be in a condition that it could be updated by the transmitted patch message and (2) an already updated mobile unit cannot constitute the claimed second mobile unit. *Id.* at *4. Given the language of claim 1 and dependent claim 14, the Board concluded that "[t]here is nothing in claim 1 that excludes a second mobile unit that has already created patched

operating code from the at least one patch message from meeting the claim 1 wherein limitation." *Id.*

Second, the Board found that Sugita discloses the wherein limitation in two ways: transmitting update information by group ID and by individual ID. *Id.* at *7–8. As to transmitting by group ID, the Board found that Sugita describes sending update information to units belonging in a specific group. *Id.* at *7. Accordingly, terminals in a specified group receive the update, while terminals not in the specified group do not. *Id.* The Board gave substantial weight to the testimony of Microsoft's expert, Dr. White, that "update information sent to the one or more mobile terminals using the group ID for the first group unit would not be meant for the one or more mobile terminals in the second group unit." *Id.* As to transmitting by individual ID, the Board found that Sugita discloses transmitting update information one unit at a time, to a non-updated mobile terminal on its target list and not to an already updated terminal or a subsequently listed terminal. *Id.* at *8.

In *Samsung I*, the Board reached the same conclusions, rejecting Iron Oak's same arguments about the construction of the wherein limitation and finding that Sugita anticipates claim 1 of the '275 patent. 2020 WL 633816, at *4, 7–9. The Board gave substantial weight to the testimony of Samsung's expert, Dr. Bederson, which was substantively similar to that of Microsoft's Dr. White. *See id.* at *7. The Board additionally found that Sugita transmits patch messages to a group unit and not a different group unit "to avoid the problem identified by Sugita of overloading the network." *Id.* at *8. Separately, the Board held that claim 1 was obvious over a combination of two other prior art references: Australian Patent Application No. 77395/91 ("Ballard") and Japanese Patent Application No. 05-66937 ("Shimizu"). *Id.* at *12.

Finally, in *Samsung II*, the Board held that claim 1 of the '275 patent was obvious over a combination of two further prior art references: U.S. Patent No. 5,619,412 ("Hapka") and U.S. Patent No. 5,442,553 ("Parrillo").

Iron Oak timely appealed to this court. We have jurisdiction pursuant to 28 U.S.C. § 1295(a)(4)(A).

II. DISCUSSION

We review the Board's factual findings for substantial evidence and its legal determinations *de novo*. *In re Stepan Co.*, 868 F.3d 1342, 1345 (Fed. Cir. 2017). A finding is supported by substantial evidence if a reasonable mind might accept the evidence to support the finding. *Nobel Biocare Servs. AG v. Instradent USA, Inc.*, 903 F.3d 1365, 1374 (Fed. Cir. 2018).

Claim construction is a question of law with underlying questions of fact. *Wasica Finance GmbH v. Continental Automotive Sys., Inc.*, 853 F.3d 1272, 1278 (Fed. Cir. 2017). Where the intrinsic record fully governs the proper construction of a term, we review the Board's claim construction *de novo*. *Id.* Anticipation is a question of fact. *Nobel Biocare*, 903 F.3d at 1375.

On appeal, Iron Oak argues that the Board erroneously construed the second mobile unit and wherein limitations to include a second mobile unit that has already created patched operating code from the patch message transmitted to the first mobile unit. Iron Oak also challenges the Board's anticipation and obviousness determinations. To the extent necessary to a resolution of these appeals, we address each of Iron Oak's arguments in turn.

A. Claim Construction

The Board correctly construed the wherein limitation not to contain the temporal restriction that Iron Oak seeks. At institution, the Board construed the wherein limitation such that the "manager host is further operable to decide

which specific mobile unit to send the at least one discrete patch message to before beginning transmission" of the message. *Microsoft*, 2020 WL 633707, at *3. None of the parties argued that the Board's interpretation was improper. *Id.* at *3; *Samsung I*, 2020 WL 633816, at *4; *Samsung II*, 2020 WL 633822, at *3. Instead, Iron Oak argued that the second mobile unit must be in a condition where its current operating code could be updated by the patch message sent to the first mobile unit. Conversely, a mobile unit whose operating code had already been updated by the patch message could not constitute the claimed second mobile unit. This is because, according to Iron Oak, at the time of transmission of the patch message, both mobile units must be operable to create patched operating code from their current operating code.

The Board rejected Iron Oak's argument as unsupported by the evidence. *Microsoft*, 2020 WL 633707, at *4. It explained, "[t]he claim language does not specify when the mobile units must be 'operable to create patched operating code' from current operating code, only that they are 'operable to' do so." *Id.* Moreover, the Board found nothing in claim 1 that excludes a second mobile unit that has already created patched operating code using the transmitted patch message from meeting the wherein limitation. *Id.* According to the Board, "[a] second mobile unit that is capable of creating patched operating code in some circumstances still satisfies the claim." *Id.*

We agree. Claim 1 recites a second mobile unit "operable to receive the at least one discrete patch message" and "further operable to create patched operating code by merging the at least one patch with current operating code located in the second mobile unit and to switch execution to the patched operating code." '275 patent, col. 13, ll. 44–49. Claim 1 also recites a manager host "further operable to address the at least one discrete patch message such that the at least one discrete patch message is transmitted to the first mobile unit but not to the second mobile unit." *Id.*

at col. 13, ll. 50–53. Neither of these limitations require the second mobile unit to be able to create patched operating code at the time the manager host addresses the patch message to the first mobile unit. Nor do these limitations exclude a second mobile unit that has already created patched operating code from the discrete patch message. In sum, there is no textual basis for Iron Oak's reading of claim 1.

Iron Oak's arguments based on the claim language, the Board's construction, dependent claims, and the specification are unpersuasive.

First, Iron Oak argues that "*the* at least one discrete patch message" and "*the* at least one patch" in the second mobile unit and wherein limitations require the second mobile unit to be operable to create patched operating code when the manager host transmits the patch message to the first mobile unit. *See id.* at col. 13, ll. 44–53 (emphases added). Not so. The claim language plainly requires the second mobile unit to be operable to create patched operating code from the patch message that the manager host sends to the first mobile unit. But, critically, the second mobile unit and wherein limitations do not impose a time frame during which the second mobile unit must be operable to create patched operating code from the transmitted patch message. The second mobile unit simply must have the capability of creating patched operating code with *the* patch message that the manager host transmits, and the manager host simply must have the capability of addressing the patch message to one mobile unit and not another.

Second, Iron Oak argues that the Board's construction renders meaningless the requirement that the second mobile unit be "further operable to create patched operating code by merging the at least one patch with current operating code located in the second mobile unit." *See id.* at col. 13, ll. 45–49. We disagree. This language requires the second mobile unit to have the capability of creating

patched operating code from the patch that is transmitted to the first mobile unit. It does not require the second mobile unit to have such capability *after* the manager host transmits the patch message to the first mobile unit. For this reason, it is irrelevant whether an already updated mobile unit can "thereafter" or "again" incorporate the transmitted patch into its operating code. *See* Appellant's Br. 22–23. Indeed, we agree with the Board that a mobile unit that has created patched operating code from the patch message transmitted to the first mobile unit is clearly operable to create patched operating code from the transmitted message, as required by the second mobile unit limitation. *See Microsoft*, 2020 WL 633707, at *4.

Third, Iron Oak argues that the Board's construction of the wherein limitation recognizes the temporal requirements of claim 1. This argument also lacks merit. The Board's construction recognizes only that the manager host *decides* to which mobile units to send the patch message before *transmitting* the message.[2] But this sequence does not implicate whether the second mobile unit (to which the manager host did not send the message) must be able to create patched operating code after the manager host transmits the message to the first mobile unit. It simply requires the manager host to decide to which mobile units to send the at least one patch message, without constraining the reasons why the manager host does not send the patch message to the second mobile unit.

---

[2] We reject any attempt by Iron Oak to stretch the Board's claim construction to require a decision to send update information to the first mobile unit and a separate decision not to send update information to the second mobile unit. The Board required the manager host only to decide to which mobile units to send the patch message. We see no basis in the Board's construction or the claim language to require a separate decision not to send.

Fourth, Iron Oak argues that dependent claim 14 supports its construction of claim 1. According to Iron Oak, because dependent claim 14 requires the manager host to be operable to address the patch message to both the first and second mobile units, claim 1 necessarily requires the second mobile unit to be operable to create patched operating code at the time the manager host transmits the patch message. We are unpersuaded. Claim 14's *narrower* scope—requiring both mobile units to be non-updated—cannot serve to constrain claim 1's *broader* scope—which only requires the first mobile unit to be non-updated. *See Phillips v. AWH Corp.*, 415 F.3d 1303, 1314–15 (Fed. Cir. 2005) (en banc) ("[T]he presence of a dependent claim that adds a particular limitation gives rise to a presumption that the limitation in question is not present in the independent claim.").

Fifth, Iron Oak argues that Figure 5 and its accompanying text requires the second mobile unit to be operable to create patched operating code from the transmitted patch message when the manager host transmits that message to the first mobile unit. This argument fails. Figure 5 and its accompanying text relate to a method of remote patching operating code and not a system. Moreover, even if Figure 5 contemplates checking a mobile unit's operability to create patched operating code after receiving a patch message, the claim language of claim 1 does not. *See SuperGuide Corp. v. DirecTV Enters., Inc.*, 358 F.3d 870, 875 (Fed. Cir. 2004) ("Though understanding the claim language may be aided by the explanations contained in the written description, it is important not to import into a claim limitations that are not a part of the claim.").

Finally, Iron Oak argues that the Board erroneously relied on *ParkerVision, Inc. v. Qualcomm Inc.*, 903 F.3d 1354 (Fed. Cir. 2018). We see no error. In *ParkerVision*, we held that "a prior art reference may anticipate or render obvious an apparatus claim—depending on the claim language—if the reference discloses an apparatus that is

reasonably capable of operating so as to meet the claim limitations, even if it does not meet the claim limitations in all modes of operation." *Id.* at 1361. The Board correctly relied on *ParkerVision* to find that a second mobile unit that is capable of creating patched operating code in some circumstances—*i.e.*, an already updated second mobile unit—still satisfies claim 1 of the '275 patent. *See Microsoft*, 2020 WL 633707, at *4. We find no basis for Iron Oak's contention that the second mobile unit's operability to create patched code is a "structure" and are unpersuaded by its attempt to distinguish the case. *See* Appellant's Br. 25. For these reasons, we hold that the Board correctly rejected the temporal restrictions that Iron Oak seeks.

B. Invalidity

Substantial evidence supports the Board's conclusion that Sugita anticipates claim 1 of the '275 patent. A prior art reference must disclose every limitation of the claimed invention, either explicitly or inherently, to anticipate. *Telemac Cellular Corp. v. Topp Telecom, Inc.*, 247 F.3d 1316, 1327 (Fed. Cir. 2001).

Iron Oak's sole challenge to the Board's findings on Sugita is that Sugita does not disclose the wherein limitation. The Board found that Sugita discloses this limitation in two ways. First, Sugita transmits update information to units belonging to a specific group. *Microsoft*, 2020 WL 633707, at *7. In this way, terminals in a specified group (*i.e.*, a first mobile unit) receive the update, while units not in the specified group (*i.e.*, a second mobile unit) do not. *Id.* Second, Sugita describes performing updates one unit at a time. *Id.* at *8. In other words, Sugita's base station transmits update information to an individual terminal on the list of update targets (*i.e.*, a first mobile unit) and not to a successfully updated terminal or a terminal later on the list (*i.e.*, a second mobile unit). *Id.*

Substantial evidence supports the Board's finding that Sugita's use of group ID to transmit update information

teaches the wherein limitation. Sugita explains that, in the initial stage, its base station transmits the update information to each mobile communications terminal "based on the group ID of the group unit (all units, or *the unit addresses belonging to a specific group*)." J.A. 390 (¶ 13) (emphasis added). Sugita reiterates that, "in the initial stage of updating, multiple communication terminals are updated *in group units*." J.A. 391 (¶ 15) (emphasis added); *accord* J.A. 393 (¶ 49). According to Dr. White, a skilled artisan would recognize that this disclosure of multiple "group units" means that "one or more mobile terminals belong to different group units." J.A. 384 (¶ 151). Dr. White further testified that "update information sent to the one or more mobile terminals using the group ID for the first group unit would not be meant for the one or more mobile terminals in the second group unit." *Id.* He concluded that update information transmitted to the first group ID is "transmitted to the first mobile unit but not the second mobile unit," as claimed. *Id.* In view of Sugita's disclosures and Dr. White's testimony, the Board reasonably found that Sugita teaches the wherein limitation.[3]

We see no reversible error in the Board's analysis. First, Iron Oak argues that "Sugita teaches an all or nothing system and method," where "every mobile terminal capable of being updated will be sent the update." Appellant's Br. 26–27. But Iron Oak's argument contradicts Sugita's express disclosure that it sends update information to terminals "based on the group ID of the group unit (all units, or *the unit addresses belonging to a specific group*)." J.A. 390 (¶ 13) (emphasis added). Sugita plainly

---

[3] Substantial evidence similarly supports the Board's finding in *Samsung I* that Sugita anticipates claim 1 in view of Sugita's disclosures and Dr. Bederson's substantially similar testimony. 2020 WL 633816, at *7–9; *see* J.A. 895 (¶ 101), 1163 (¶ 13).

contemplates updating mobile communications terminals "in group units." J.A. 391, 393 (¶¶ 15, 49). It is therefore irrelevant that Sugita lists all mobile terminals targeted for update or that there is no terminal "m_n" that is excluded from the list. *See* Appellant's Br. 27. Iron Oak at most shows that all mobile terminals eventually receive update information in the initial stage. But such a showing does not undercut the Board's finding that Sugita can send update information to terminals on its target list in a specific group and not to terminals on its target list in another group.[4] *See Microsoft*, 2020 WL 633707, at *7–8.

Second, Iron Oak misconstrues Dr. White's testimony that update information sent to terminals in a first group unit "would not be meant for" terminals in a second group unit. *See* Appellant's Br. 29–30, 29 n.5. According to Iron Oak, Sugita's update information is meant for terminals in both group units because all of the terminals are on Sugita's list of terminals to be updated. But the Board reasonably understood Dr. White's testimony to mean "update information transmitted to the first group ID is 'transmitted to the first mobile unit but not to the second mobile unit.'" *Microsoft*, 2020 WL 633707, at *7. In this way, the update information is "meant for"—*i.e.*, transmitted to—the first group unit and not the second group unit.

For these reasons, we hold that substantial evidence supports the Board's finding that Sugita's use of group ID to update terminals teaches the claimed wherein

---

[4] Iron Oak contends that the Board confused the mobile units' operability to receive updates with the manager host's operability to decide to which units to send the updates. The Board did not misunderstand claim 1. Rather, the fact that terminals in a specific group (and not terminals in another group) receive the update demonstrates that Sugita's base station decides to which mobile units to send the updates.

limitation. We therefore need not reach the Board's finding on Sugita's use of individual IDs. And because we hold that substantial evidence supports the Board's finding that Sugita anticipates claim 1 of the '275 patent, we need not reach the Board's conclusions on obviousness over Ballard and Shimizu in *Samsung I* or obviousness over Hapka and Parrillo in *Samsung II*.

III. CONCLUSION

We have considered Iron Oak's remaining arguments and find them unpersuasive. For the reasons discussed above, we *affirm*.

AFFIRMED