NOTE: This disposition is nonprecedential.

United States Court of Appeals for the Federal Circuit

SWARM TECHNOLOGY LLC,
Appellant

v.

AMAZON.COM, INC., AMAZON WEB SERVICES, INC.,
Cross-Appellants

2023-2323, 2024-1095

Appeals from the United States Patent and Trademark Office, Patent Trial and Appeal Board in Nos. IPR2022-00283, IPR2022-00633.

Decided: June 30, 2025

MEREDITH LEIGH MARTIN ADDY, AddyHart P.C., Atlanta, GA, argued for appellant. Also represented by DANIEL JOSEPH ANDERSON, CHRISTINE N. JONES, MICHAEL K. KELLY, Newman Jones PLLC, Scottsdale, AZ.

ADAM MICHAEL GREENFIELD, White & Case LLP, Washington, DC, argued for cross-appellants. Also represented by MARK S. DAVIES; HALLIE ELIZABETH KIERNAN, New York, NY.

Before TARANTO, STOLL, and STARK, *Circuit Judges*.

STARK, *Circuit Judge*.

In Final Written Decisions ("FWD") issued by the Patent Trial and Appeal Board ("Board") in two related inter partes review ("IPR") proceedings, the Board found that Petitioner Amazon.com Inc. ("Amazon") proved unpatentable some, but not all, claims of two patents owned by Swarm Technology LLC ("Swarm"). Swarm appeals the findings of unpatentability while Amazon cross-appeals the findings that it failed to prove some claims unpatentable. Swarm also appeals the denial of its contingent motion to amend to add substitute claims. For the reasons that follow, we affirm the Board in full.

I

Swarm's U.S. Patent No. 9,852,004 ("'004 patent") is entitled "System and Method for Parallel Processing Using Dynamically Configurable Proactive Co-Processing Cells." J.A. 143. Its U.S. Patent No. 10,592,275 ("'275 patent") is entitled "System and Method for Swarm Collaborative Intelligence Using Dynamically Configurable Proactive Autonomous Agents." J.A. 159. Both the '004 and '275 patents share a nearly identical specification and claim priority to U.S. Patent 9,146,777, which was filed in 2013. In general, both patents are directed to parallel multiprocessing computer architecture used for completing computing tasks.

In parallel or multi-core processing, a central processing unit ("CPU") breaks down large computational tasks into individual blocks of computations; the CPU then distributes the tasks among two or more processors. More particularly, in the '004 patent, a CPU places tasks into a task pool. Co-processors then retrieve a task, complete it, notify the task pool the task was completed, and then "ping[] the task pool until another task becomes available." J.A. 151 ('004 patent 2:16-17). Each co-processor can include "an agent that interrogates the task pool to seek a task to perform." J.A. 151 ('004 patent 2:23-24). "[T]he term agent refers to a software module, analogous to a network packet, associated with a co-processor that interacts with the task pool to thereby obtain available tasks which are appropriate for that co-processor cell." J.A. 152 ('004 patent 3:13-16). In one embodiment, the agent "is generally analogous to a data frame in the networking sense, in that an agent may be equipped with a source address, a destination address, and a payload." J.A. 154 ('004 patent 8:30-34).

Claim 3 of the '004 patent is representative of the issues presented in this appeal and recites:

> A processing system, comprising:
>
> a task pool;
>
> a controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks;
>
> a first co-processor configured to successively: retrieve a first task from the task pool; deliver the first task to the first co-processor; process the first task; generate first resulting data; and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller; and
>
> a second co-processor configured to successively: retrieve a second task from the task pool; deliver the second task to the second co-processor; process the second task; generate second resulting data; and update the task pool to reflect completion of the second task, all without any communication between the second co-processor and the controller;

wherein:

the processing system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the processing system *on a plug-and-play basis without any communication with the controller*;

the first task includes indicia of a first task type, the first co-processor is configured to perform tasks of the first type, *and the first agent is configured to search the task pool for a task of the first type*;

the second task includes indicia of a second task type, the second co-processor is configured to perform tasks of the second type, *and the second agent is configured to search the task pool for a task of the second type*;

the first co-processor includes a first agent comprising a first source address, a first destination address, and a first payload; and

the second co-processor includes a second agent comprising a second source address, a second destination address, and a second payload;

and further wherein:

when the first agent is retrieving the first task from the task pool, the first source address corresponds to an address associated with the first co-processor, the first destination address corresponds to an address associated with the task pool, and the first payload includes a first function which the first co-processor is configured to perform;

when the first agent is returning from the task pool, the first source address is the task pool's address, the first destination address is the first co-processor's address, and the first payload includes a descriptor of the first task;

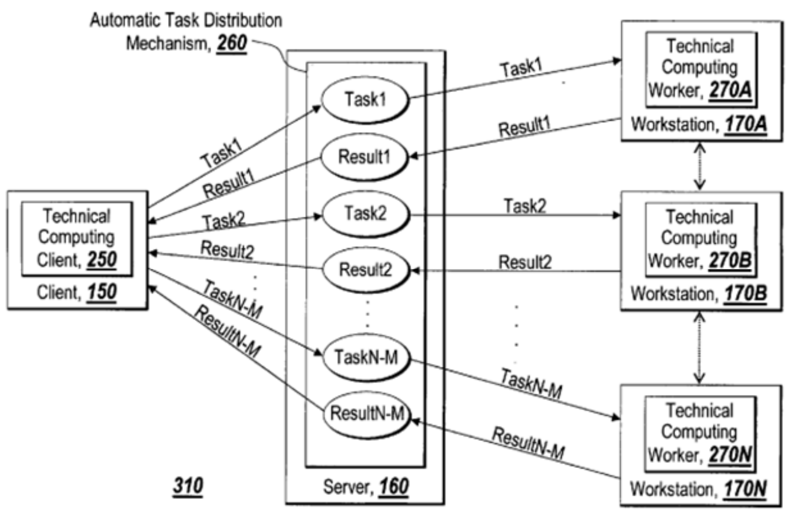
> when the second agent is retrieving the second task from the task pool, the second source address corresponds to an address associated with the second co-processor, the second destination address corresponds to an address associated with the task pool, and the second payload includes a second function which the second co-processor is configured to perform; and
>
> when the second agent is returning from the task pool, the second source address is the task pool's address, the second destination address is the second co-processor's address, and the second payload includes a descriptor of the second task.

J.A. 157-58 ('004 patent 14:42-15:36) (emphasis added).

The '275 patent contains a similar limitation in independent claim 6, which recites a system within which a "first agent is configured to search the task pool for a task of the first type" and a "second agent is configured to search the task pool for a task of the second type." J.A. 174 ('275 patent 15:43-44, 15:47-48). The claims of the '275 patent additionally require that the co-processors are configured to "proactively retrieve" a task from the task pool. *See* J.A. 173-74. According to the specification, the term "proactive" "suggests that each co-processor may be configured (e.g., programmed) to periodically send an agent to monitor the task pool for available tasks appropriate to that co-processor." J.A. 151 ('004 patent 2:40-43).

On Amazon's petitions, the Board instituted IPRs challenging all claims of the '004 and '275 patents. The grounds for the petition included that certain claims are obvious over United States Patent Application Publication No. 2007/0124363 ("Lurie") and a 1980 publication by Digital Equipment Corporation, Intel, and Xerox entitled "The Ethernet-A Local Area Network, Data Link Layer and Physical Layer Specifications" ("Ethernet"). J.A. 350-53, 1219, 1286, 4275, 4278-79.

Lurie "allows the use of instrument-based distributed system[s] on a network to conduct the job and facilitate decreasing the time for executing the job." J.A. 1247 (¶ 5). As shown below in Lurie's Figure 3B, Lurie allows for the distribution of tasks in an automatic distribution mode where a technical computing client [250] puts tasks into an automatic task distribution mechanism [260], i.e., a task pool. Lurie's workers [270] take a task from the task pool, perform the task, and then return the result of the task to the task pool. J.A. 1253 (¶ 72).



J.A. 1226 (Figure 3B).

The Ethernet publication deals primarily with how devices are distributed and connected on a local area network ("LAN"), disclosing a standard for establishing a network that provides "high speed data exchange among computers within a . . . geographic area." J.A. 1296. Ethernet is "intended primarily for use in such areas as . . . distributed data processing." J.A. 1296. Data is sent, via data frames, through the LAN, with a protocol for including information for the destination address, source address, type, data, and error checking sequence.

The Board found that Lurie in view of Ethernet, along with other references not at issue on appeal, taught the limitations of all claims (1-12) of the ’004 patent and rendered each of them obvious.[1] Swarm moved to amend to add substitute claims 13 and 14. The Board denied the motion because the limitation being added in the proposed substitute claims – “dynamically accept[ing]” an “additional co-processor into the processing system on a plug-and-play basis without any communication with the controller” – lacked support in the original application. J.A. 78 (alteration in original; emphasis omitted). The Board further determined that the substitute claims were directed to nonpatentable subject matter under 35 U.S.C. § 101. In the IPR concerning the ’275 patent, the Board construed the term “proactively” and, applying that construction, found Amazon failed to prove that any of the challenged claims (1-17) were unpatentable.

Swarm timely appealed the Board’s FWD in connection with the ’004 patent IPR and Amazon did so in connection with the ’275 patent IPR. We have jurisdiction under 28 U.S.C. § 1295(a)(4)(A).

II

“The Board’s claim constructions . . . are determinations of law reviewed de novo where based on intrinsic evidence, with any Board findings about facts extrinsic to the patent record reviewed for substantial-evidence support.” *St. Jude Med., LLC v. Snyders Heart Valve LLC*, 977 F.3d 1232, 1238 (Fed. Cir. 2020). “Substantial evidence is such relevant evidence as a reasonable mind might accept as adequate to support a conclusion.” *IOENGINE, LLC v. Ingenico Inc.*, 100 F.4th 1395, 1402 (Fed. Cir. 2024) (internal quotation marks omitted). “[T]he possibility of drawing

---

[1] Swarm does not contest the Board’s determination that claim 1 of the ’004 patent is unpatentable.

two inconsistent conclusions from the evidence does not prevent an administrative agency's finding from being supported by substantial evidence." *Consolo v. Fed. Mar. Comm'n*, 383 U.S. 607, 620 (1966).

"The ultimate question of obviousness is a legal question that we review de novo with underlying factual findings that we review for substantial evidence." *Roku, Inc. v. Universal Elecs., Inc.*, 63 F.4th 1319, 1324 (Fed. Cir. 2023). The underlying factual findings include: the scope and content of the prior art; the differences between the prior art and the claims at issue; the level of ordinary skill in the pertinent art; along with secondary considerations such as commercial success, long felt but unsolved needs, and failure of others. *See Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966).

Written description under 35 U.S.C. § 112 is a question of fact reviewed for substantial evidence. *Blue Calypso, LLC v. Groupon, Inc.*, 815 F.3d 1331, 1344 (Fed. Cir. 2016).

III

Swarm appeals the Board's finding that the combination of Lurie and Ethernet discloses the "agent . . . configured to search the task pool" limitation and its related conclusion that Lurie and Ethernet render claims 2-12 of the '004 patent obvious. Swarm also appeals the Board's denial of its motion to amend to add substitute claims 13 and 14 to the '004 patent. We address both contentions below before turning to Amazon's cross appeal of the Board's conclusion that claims 1-17 of the '275 patent were not unpatentable.

A

The Board had substantial evidence to support its finding that the claim limitation "agent . . . configured to search the task pool," which appears in claims 2-12 of the '004 patent and claims 3 and 6-10 of the '275 patent, is disclosed by the combination of Lurie and Ethernet. The Board explained that the "data frames sent by the [technical computing workers] in Lurie as modified by Ethernet . . . act as agents which search the [automatic task distribution] [m]echanism [of Lurie] for tasks that the [technical computing workers] can perform." J.A. 48. The Board concluded it would have been obvious to modify Lurie to use Ethernet's "data frame as an agent," resulting in the worker of Lurie searching the task pool via that agent, as Lurie's worker "selects or determines which tasks it will perform" and Lurie's "system can determine or select the technical comput[ing] worker . . . to work on a task by operational and/or performance characteristics of the technical computing worker." J.A. 39-40 (citing Lurie ¶ 81; emphasis omitted). The Board concluded that because Lurie teaches that the workers are not homogenous, and instead, can differ in capabilities, a skilled artisan would have considered it obvious to allow an agent to rely on a task description specifying the necessary capabilities a worker needs to have to complete a type of task in order to search for the corresponding types of tasks the worker can perform. J.A. 39, 41. The Board found a "person having ordinary skill in the art would have understood that this [modification] would allow the agent to search for tasks that the [worker] can perform . . . [to] maximize the efficiency of the system." J.A. 41.

Swarm levels several attacks on the Board's analysis, but none persuades us that the Board lacked substantial evidence for its findings. First, Swarm argues that the Board erred by implicitly construing "search" in a manner inconsistent with its plain meaning, which in Swarm's view "requires the ability to evaluate and choose among different tasks within the task pool in order to selectively retrieve one with an appropriate task type." Open. Br. at 18-19. Neither party asked the Board to construe the claim term "search," and the Board did not explicitly do so; nor do we see anything in the Board's understanding of the term that causes us to believe its view of the term was materially different than Swarm's proposed plain and ordinary meaning. Swarm's claim construction contention, then, lacks merit.

Swarm next asserts that Lurie teaches workers being assigned tasks by the automatic task distribution mechanism, not that the workers themselves search through a pool of tasks to find and select one matching their operational and performance characteristics. In support of its contention, Swarm highlights paragraph 81 of Lurie, on which the Board relied, which states in pertinent part: "Additionally, in any of the embodiments depicted in FIGS. 3A-3D, the system can determine or select the technical computer worker 270A-270N to work on a task by operational and/or performance characteristics of the technical computing worker 270A-270N and/or workstation 170A-170N." J.A. 1255 (Lurie ¶ 81) (quoted in Open. Br. at 23 and by Board at J.A. 40). Swarm contends that in Lurie's "system," the automatic task distribution mechanism, not the worker, does the selecting.

The Board carefully considered this argument and rejected it. In doing so, the Board looked at the entirety of Lurie's disclosure, including the portion on which Swarm now relies on appeal, and found that its teachings, in combination with the teachings of Ethernet, render the disputed claim limitation obvious. J.A. 40-41 ("Read in context with another portion of the [Lurie] Specification, it is clear that it is the [technical computing workers] that are selecting the tasks.").

Substantial evidence, including Lurie itself, supports this finding. For instance, Lurie states that "[t]he technical computing client . . . does not need to know the specifics of the technical computing worker. . . . As such, the automatic task distribution mechanism provides a level of indirection between the technical computing client . . . and the technical computing worker," J.A. 40 (citing J.A. 1251 (Lurie ¶ 61)), and "the technical computing client . . . is not required to have any specific knowledge of the technical computing workers, . . . e.g., the name of the workstation running a technical computing worker . . ., or the availability of the technical computing worker . . . to perform technical computing of a task," J.A. 40-41 (citing J.A. 1253 (Lurie ¶ 71)). In this context, the Board reasonably concluded that paragraph 81's "system" includes workers, that can "determine or select" if they should work on a task, based on "operational and/or performance characteristics." While perhaps other conclusions might reasonably be drawn from these teachings, the Board's conclusion – that Lurie teaches an agent selecting a task type that matches some feature or capability of the co-processor – is certainly a reasonable one, and hence is supported by substantial evidence.

Attacking the combination of Lurie and Ethernet, Swarm faults the Board for relying on the allegedly inaccurate declaration of Amazon's expert, Dr. David Lowenthal, to fill "gaps" in Lurie, and for providing a merely conclusory, insufficient analysis of motivation to combine. We are not persuaded. Dr. Lowenthal opined that Ethernet teaches a "standard protocol;" that is, a "well known and successful method[] of implementing networked communications over a LAN." J.A. 787 (¶ 213). The Board credited this testimony to find that a skilled artisan "would have sufficient reason to combine" Lurie and Ethernet "with a reasonable expectation of success." J.A. 45, 55; *see also KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (explaining combination is obvious when it "simply arranges old elements with each performing the same function it had been known to perform and yields no more than one would expect") (internal quotation marks omitted). The Board found that Ethernet provides a way of working a standard LAN setup, including data frames that correspond to the claimed agents, and Lurie is disclosed as running on a LAN. Reasons to implement Ethernet's LAN on Lurie, according to Dr. Lowenthal, include how ubiquitous and well-known LAN protocol was, that Ethernet also addressed distributed data processing, and that the teachings of the Ethernet reference would have ensured compatibility and interoperability between the multiple devices in Lurie's system. *See, e.g.*, J.A. 768-69 (¶¶ 177-78). Dr. Lowenthal's extensive reasons to implement Ethernet's LAN are not conclusory and, as the Board observed, Swarm "does not address separately" Dr. Lowenthal's reasoning. J.A. 55; *see also* J.A. 42, 53, 65 (Board finding Dr. Lowenthal's testimony "persuasive," not conclusory). Thus, in context, we can reasonably discern the Board's reasoning, which substantially tracks that of Dr. Lowenthal, as to why a skilled artisan would have been motivated to combine Lurie and Ethernet and would have had a reasonable expectation of success in doing so. *See In re Nuvasive, Inc.*, 842 F.3d 1376, 1382 (Fed. Cir. 2016) ("[W]e may affirm the [Board's] findings if we may reasonably discern that it followed a proper path, even if that path is less than perfectly clear.") (internal quotation marks omitted).

B

Swarm next challenges the Board's denial of its motion to amend by replacing claims 1 and 2 with substitute claims 13 and 14 in the '004 patent, because the original specification did not provide adequate written description support for the new limitation Swarm sought to add in these substitute claims. *See* J.A. 81 ("Various sections of the specification . . . simply discuss adding processors or the plug-and-play feature [but] none of the paragraphs describe another processor added 'on a plug-and-play basis without any communication with the controller.'"); *see also generally* J.A. 79-84. In particular, the Board found there was insufficient support for the limitation "wherein the processing system is configured to dynamically accept the first autonomous co-processor, the second autonomous co-processor, and an additional autonomous co-processor into the processing system on a plug-and-play basis *without any communication with the controller*." J.A. 1842 (emphasis added).

Swarm criticizes the Board for having such a broad understanding of this limitation that it excludes direct and indirect communication. Open. Br. at 56 (citing J.A. 84). According to Swarm, the other words of the claim, including the recitation of "a controller configured to populate the task pool with . . . tasks," instead require that the component retrieving a task be in indirect communication with the controller via the task pool. Thus, requiring the claims to be practiced without any communication, including without this type of indirect communication, "cannot be correct" as it renders the claims "facially nonsensical." Open. Br. at 58 (quoting *Becton, Dickinson & Co. v. Tyco Healthcare Grp., LP*, 616 F.3d 1249, 1255 (Fed. Cir. 2010)). To Swarm, the "correct construction of 'without any communication' . . . [is] 'without any *direct* communication.'" Open. Br. at 60.

We disagree with Swarm and agree with the Board. The plain and ordinary meaning of "any" includes "of all types," so "any communication" includes "all types" of communication and, specifically, "direct and indirect communication." It follows that to be "without any communication with the controller" means to be without direct and without indirect communication with the controller. Moreover, the specification expressly distinguishes between direct and indirect communication. *See, e.g.*, J.A. 151 ('004 patent 2:17-19) ("[T]he CPU communicates directly with the task pool[] and communicates indirectly with the co-processors through the task pool."). This evidences the inventor's understanding of "any communication" as including both types of communication. J.A. 84; *see also generally* Amazon's Resp. Br. at 45 (contending Swarm argues, for first time on appeal, "'without any communication' actually allows some communication") (emphasis omitted).

There is, however, nothing in the specification to show a person of ordinary skill in the art that the inventor possessed an embodiment in which a co-processor is dynamically added into the system without any communication –

direct or indirect – with the controller. Swarm attempts to rely on intrinsic evidence arguably supporting a view that the controller may generally communicate indirectly with co-processors during operation. But this purported evidence does not compel a conclusion that one of skill in the art would understand "without any communication" to also allow indirect communications during the dynamic process of adding a new co-processor to the system. Hence, substantial evidence supports the Board's finding that the specification fails to provide adequate written description support for the substitute claims and their negative limitation. *See Novartis Pharms. Corp. v. Accord Healthcare, Inc.*, 38 F.4th 1013, 1016 (Fed. Cir. 2022) ("For negative claim limitations . . . there is adequate written description when, for example, the specification describes a reason to exclude the relevant [element].") (internal quotation marks omitted; second alteration in original).

The specification support Swarm identifies as purportedly providing sufficient written description at best shows the inventor possessed embodiments in which there was no direct communication between the co-processors and the controller. That is, Swarm's written description argument is bound up with its claim construction position, which we have rejected. We affirm the Board's denial of Swarm's motion to amend.[2]

IV

We now turn to Amazon's cross-appeal, which is based on what Amazon contends is the Board's incorrect construction of "proactively" as that term is used in the claims

---

[2] Given our conclusion with respect to written description, we need not consider Amazon's additional arguments for affirmance or Swarm's challenge to the Board's conclusion that its substitute claims are directed to patent ineligible subject matter under 35 U.S.C. § 101.

of Swarm's '275 patent. Amazon argued before the Board that the term "proactively" encompasses a scenario where "each co-processor may be configured (e.g., programmed) to periodically send an agent to monitor the task pool for available tasks appropriate to that co-processor." J.A. 117 (citing J.A. 7098 (quoting J.A. 167 ('275 patent 2:48-51)). The Board, by contrast, construed "proactively retrieving a task" to mean "initiating change rather than reacting to events (i.e. not being told to act or not reactionary)." J.A. 122. Applying that construction, it found that Lurie failed to teach a proactive embodiment because Lurie required an initial kickoff signal, sent from the controller to the co-processor prior to the first task being retrieved. J.A. 137 ("Amazon fails to appreciate that each [technical computer worker] . . . still requires a kickoff notification before it begins retrieving and completing tasks. . . . [B]ecause Lurie's [technical computer workers] . . . require a kickoff notification to retrieve a first task, it does not matter that they can retrieve and complete additional tasks without further notifications."). Thus, in the Board's view, Lurie's example does not disclose or render obvious proactive task retrieval by the co-processors. J.A. 136 (crediting Dr. Lowenthal's testimony that Lurie teaches initial notification to "inform[] all registered [technical computing workers] that one or more tasks are available") (internal quotation marks omitted). Simply put, Lurie's workers reacted to the kickoff signal and therefore were not proactive in retrieving tasks, even if the workers did later on return to retrieve additional tasks after completing the initially retrieved tasks.

While the Board's construction of "proactively" means the claims do not encompass scenarios in which tasks are retrieved and completed at any point following a kickoff notification, even if no worker needs to be told anything further after the kickoff, the Board identified substantial intrinsic and extrinsic evidence supporting its construction, including examples in the specification, dictionary definitions, and expert testimony. In particular, the Board's construction finds support in the specification, even in the sentence on which Amazon places most emphasis on appeal. Amazon points to the '275 patent's statement that "[t]he term proactive suggests that each co-processor may be configured (e.g., programmed) to periodically send an agent to monitor the task pool for available tasks appropriate to that co-processor." Amazon's Resp. Br. at 68-70; *see also* J.A. 167 ('275 patent 2:48-51). The Board concluded that "suggests" as used in this sentence does not reveal the patentee adopting a special definition of "proactive."

The Board also cited extrinsic evidence, particularly dictionary definitions, which define "proactive" as "anticipatory" or "the opposite of reactive," which to the Board "support applying the ordinary and customary meaning of proactively." J.A. 121 (citing Dictionary.com and the American Heritage Dictionary (5th ed. 2011)) (internal quotation marks and emphasis omitted). The Board further looked to the testimony of Dr. Lowenthal, who explained that proactively "by its plain and ordinary meaning refers to a co-processor that does not need to be told to act." J.A. 122.

Amazon complicated the situation by not proposing a specific, alternative construction of "proactively" and by leaving it unclear whether its appeal presents a challenge to the Board's claim construction or, instead, only an appeal of the Board's application of its construction, resulting in a factual finding that it contends is somehow wrong as a matter of law. *See, e.g.*, J.A. 119 (Board explaining that "[t]he problem we face here is that neither party takes a definitive position as to what exactly constitutes the ordinary and customary meaning of the claim term 'proactively.' . . . [W]e are left to speculate what the parties believe is the ordinary and customary meaning of the claim term 'proactively' in the context of the '275 patent."); Amazon's Resp. Br. at 25 (contending Board's purported error either "stems from its incorrect construction of the term 'proactively' or, at worst, misapplication of obviousness law"); *id*. at 74 (trying to explain that if court does not view issue on appeal as "a matter of claim construction [as Amazon contends] and instead believes that it was an application" of the Board's construction, "then the Board erred by misapplying the law of obviousness"). Given the Board's reliance, in part, on extrinsic evidence, *see* J.A. 121, and our deferential review of the Board's findings about that extrinsic evidence, combined with the specification support relied on by the Board, we agree with the Board's construction. Substantial evidence supports the Board's application of that construction, including the teachings of Lurie and the testimony of Amazon's own expert, Dr. Lowenthal, *see* J.A. 122 (citing J.A. 764-67, J.A. 7303).[3]

Accordingly, we affirm the Board's FWD that Amazon failed to prove the challenged claims of the '275 patent unpatentable.

V

We have considered the parties' remaining arguments and find them unpersuasive. For the reasons stated above,

---

[3] Amazon additionally argues that the Board's construction of "proactively" is wrong because it is "narrower than any usage of the term in the '275 patent" and excludes preferred embodiments expressly taught in the patent's specification. Amazon's Resp. Br. at 71. Amazon's argument assumes, without evidence, that the embodiments of the '275 patent must have "an initial kickoff notification that tasks are available before the co-processors send an agent for monitoring." *Id*. at 69. Because we find that the specification does not support this contention, and for the other reasons given by the Board and by us in this opinion, we are not persuaded that Amazon's construction is correct.

we affirm the Board's FWDs in both IPRs related to the '004 and '275 patents.

AFFIRMED