Trials@uspto.gov
571-272-7822

Paper 45
Date: February 28, 2025

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NETFLIX, INC.,
Petitioner,

v.

AVAGO TECHNOLOGIES INTERNATIONAL SALES PTE. LIMITED,
Patent Owner.

IPR2021-00431
Patent 7,457,722 B1

Before MELISSA A. HAAPALA, *Vice Chief Administrative Patent Judge*, KRISTEN L. DROESCH, and NATHAN A. ENGELS, *Administrative Patent Judges*.

ENGELS, *Administrative Patent Judge*.

JUDGMENT
Final Written Decision on Remand
Determining All Challenged Claims Unpatentable
35 U.S.C. §§ 144, 318

I. INTRODUCTION

This Final Written Decision is entered pursuant to 35 U.S.C. § 318 and pursuant to a remand from the United States Court of Appeals for the Federal Circuit in *Netflix, Inc. v. Avago Technologies International Sales PTE. Limited*, 2024 WL 3507601 (Fed. Cir. July 23, 2024).[1] For the reasons explained below, we determine, in view of the Federal Circuit's decision and the record, that Netflix, Inc. ("Petitioner") has proven by a preponderance of the evidence that the challenged claims are unpatentable.

II. BACKGROUND

Petitioner filed a Petition (Paper 2, "Pet.") requesting *inter partes* review of claims 4, 6–8, 12, 14–16, 20, and 22 of U.S. Patent No. 7,457,722 B1 (Ex. 1001, "the '722 patent"), which we granted (Paper 8). During the trial phase, Avago Technologies International Sales Pte. Limited ("Patent Owner") filed a Response (Paper 20, "PO Resp."), Petitioner filed a Reply (Paper 23, "Reply"), and Patent Owner filed a Sur-reply (Paper 25, "Sur-reply"). Each party presented oral arguments at a hearing, and a transcript of the hearing was entered into the record. Paper 31. At the conclusion of trial, we issued a Final Written Decision determining Petitioner had not proven by a preponderance of the evidence that any of the challenged claims were unpatentable. Paper 33.

Petitioner appealed our Final Written Decision to the Court of Appeals for the Federal Circuit. The Federal Circuit issued a decision vacating our Final Written Decision and remanding for further proceedings. *Netflix, Inc. v. Avago Techs. Intl Sales PTE.*, No. 2022-1936, 2024 WL 3507601 (Fed. Cir. July 23, 2024). Among other things, the Federal Circuit

[1] The Federal Circuit's slip opinion is in the record as Paper 37.

stated that on remand, the Board should "adopt a new construction that includes application instance-level, application-level, device-level, and tier-level data" and "the Board should determine whether Masters, including its Decision Review Display and Path Display, renders the 'collecting' and 'correlating' limitations obvious in view of the Board's new construction. In addition, the Board should consider Masters' Appendix J in assessing whether Masters' Path Display teaches life cycle events and the claimed correlating step." *Netflix*, 2024 WL 3507601 at *8.

After the Federal Circuit issued its mandate, with our authorization, the parties filed the following briefs: Petitioner's Opening Brief on Remand (Paper 37, "Pet. Remand Br."), Patent Owner's Response Brief on Remand (Paper 40, "PO Remand Resp."), Petitioner's Reply Brief on Remand (Paper 41, "Pet. Remand Reply"), and Patent Owner's Sur-reply Brief on Remand (Paper 42, "PO Remand Sur-reply").

A. Related Matters

The parties state that the '722 patent has been asserted in *Broadcom Corp. et al. v. Netflix, Inc.*, Case No. 3:20-cv-04677 (N.D. Cal. 2020), which was originally filed as *Broadcom Corp. et al. v. Netflix, Inc.*, Case No. 8:20-cv-00529 (C.D. Cal.). Pet. 93; Paper 18, 1.

In addition, claims 1–3, 5, 9–11, 13, 17–19, and 21 of the '722 patent are the subject of IPR2021-00045, which was also remanded to the Board by the Federal Circuit's decision.

B. Real Parties-in-Interest

Petitioner identifies Netflix, Inc. and Netflix Streaming Services, Inc. as the real parties-in-interest. Pet. 93. Patent Owner states that Avago

Technologies International Sales Pte. Limited is the real party-in-interest. Paper 6, 1.

C. The '722 Patent

Titled "Correlation of Application Instance Life Cycle Events In Performance Monitoring," the '722 patent describes monitoring the performance of instances of a software application and correlating performance data to application instance life cycle events in a distributed computing system. Ex. 1001, codes (54), (57). In a distributed computing system, multiple computers work together in a coordinated manner. Each computer, or node, may execute multiple instances of one or more applications,[2] and instances of an application may be created or destroyed as demand changes. Ex. 1001, 1:30–36. "Application instances may also migrate from node to node in response to a hardware or software failure, or in response to a load-balancing algorithm." Ex. 1001, 1:37–39. The creation, destruction, and migration of application instances are referred to as "life cycle events." Ex. 1001, 1:54–59.

The '722 patent is directed to monitoring performance in such an environment. Ex. 1001, 1:53–56. Specifically, the '722 patent describes monitoring the life cycle events undergone by each application instance and relating those life cycle events to collected performance data for each application instance. Ex. 1001, 3:35–40. The system and method of the '722 patent "collect[s] performance data for one or more application instances, detect[s] one or more instance life cycle events associated with the

[2] An application is a piece of software. If a piece of software is installed and run on a computer, the software running on the computer is an "application instance." Ex. 1001, 1:8-49, 10:2–3.

one or more application instances, correlat[es] the performance data to the one or more instance life cycle events, and stor[es] the correlated performance data." Ex. 1001, 1:59–65.

Examples of life cycle events include creation, destruction, or migration of a particular application instance; these events may occur, for example, as demand changes or in response to a load-balancing algorithm. Ex. 1001, 1:35–39, 1:58–59. Correlating performance data to such life cycle events may comprise determining the change in performance of an application as one or more application instances are created, destroyed, or migrated from one location to another. Ex. 1001, 2:3–10.

D. Illustrative Claim

Of the challenged claims, claims 6, 14, and 22 are independent claims. The challenged claims also include dependent claims (claims 4, 12, and 20) that depend from independent claims 1, 9, and 17, which were among the claims challenged in IPR2021-00045. Claim 6, reproduced below, is representative of the independent claims challenged in this proceeding.

> 6. A method, comprising:
>
> monitoring performance of an application instance that is a computer program executing on a computer system, wherein said monitoring comprises collecting performance data associated with the performance of the application instance executing on the computer system;
>
> detecting a migration event associated with the application instance, wherein the migration event comprises the application instance migrating from a first location of execution to a second location of execution;
>
> subsequent to said detecting, continuing to monitor the performance of the application instance;

correlating the performance data to the migration event; and

storing the correlated performance data.

E. Alleged Grounds of Unpatentability

Petitioner asserts that claims 4, 6–8, 12, 14–16, 20, and 22 are unpatentable on the following grounds:

Claims Challenged	35 U.S.C. §[3]	References/Basis
4, 6–8, 12, 14–16, 20, 22	103	Masters[4]

Pet. 15.

The parties additionally cite the testimony of Dr. Prashant Shenoy (Ex. 1003; Ex. 1029; Exs. 2002–2005) and Dr. David Rosenblum (Ex. 2001; Ex. 1028).

F. Summary of Masters

Masters describes a Resource Management Architecture ("RMA") that serves as a monitoring system for a distributed computing system having a plurality of hosts capable of executing multiple copies of a scalable application. Ex. 1004, 2:44–47, 4:8–12. Masters teaches "provid[ing] capabilities for monitoring system performance along with the ability to dynamically allocate and reallocate system resources as required." Ex. 1004, 4:27–30. Figure 1A of Masters is reproduced below.

[3] The Leahy-Smith America Invents Act ("AIA") included revisions to 35 U.S.C. § 103 that became effective on March 16, 2013. Because the '722 patent has a priority date before March 16, 2013, we apply the pre-AIA version of § 103. *See* Ex. 1001, code (22).
[4] U.S. Patent No. 7,051,098 B2, issued May 23, 2006 (Ex. 1004).

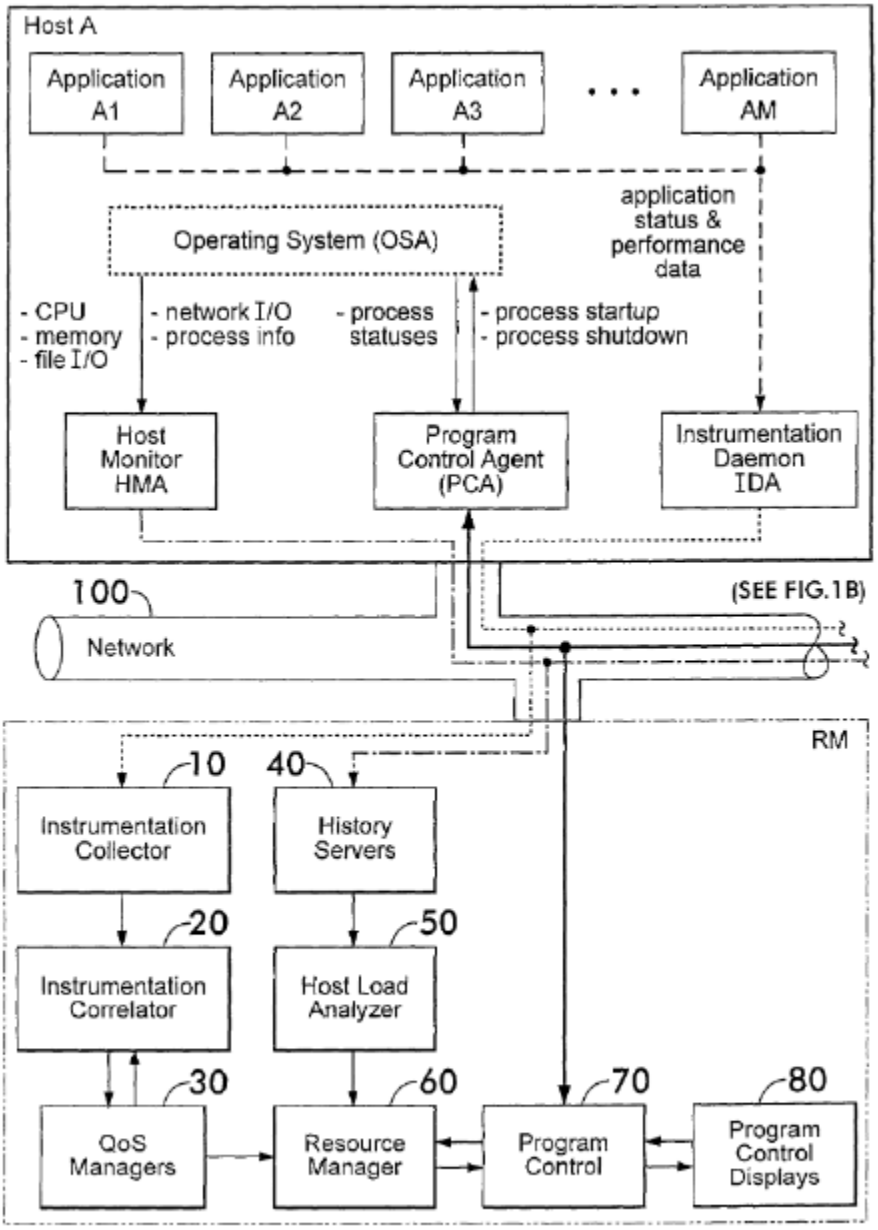


FIG.IA

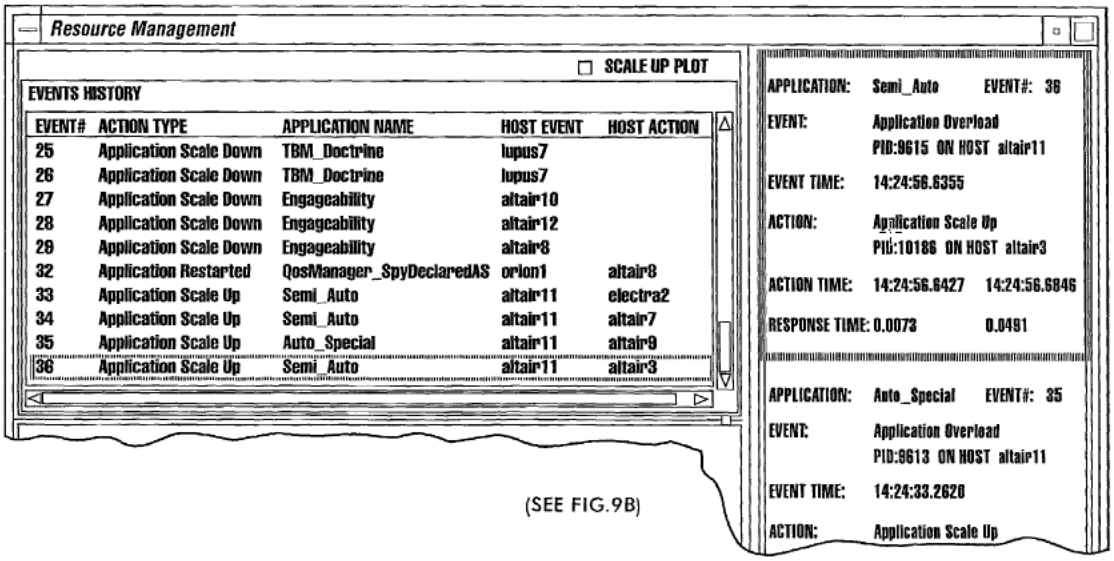
Figure 1A of Masters shows Host A executing instantiated applications A1–AM. Ex. 1004, 5:45–61. Host A also has an instrumentation daemon[5] that

[5] Masters defines a "daemon" as "[a] background process on a host or Web server . . . waiting to perform tasks." Ex. 1004, col. 4, table.

receives "application status & performance data" from applications A1–AM. Ex. 1004, 5:52–61.

The RMA, labeled "RM" in Figure 1A, includes instrumentation collector 10 that receives data from the instrumentation daemon in each host over network 100 and provides data to instrument correlator 20. Ex. 1004, 5:35–37, 62–65. Instrumentation correlator 20 provides correlation data to quality of service ("QoS") managers 30, which in turn provide information to resource manager 60. Ex. 1004, 5:65–67, 6:40–42. Resource manager 60 communicates with program controller 70, which "sends startup and shutdown orders" to the program control agents in each host. Ex. 1004, 6:35–49. Masters teaches the need for "allowing autonomous start up and shut down of application copies on host machines to accommodate changes in data processing requirements." Ex. 1004, 2:20–24.

FIG.9A



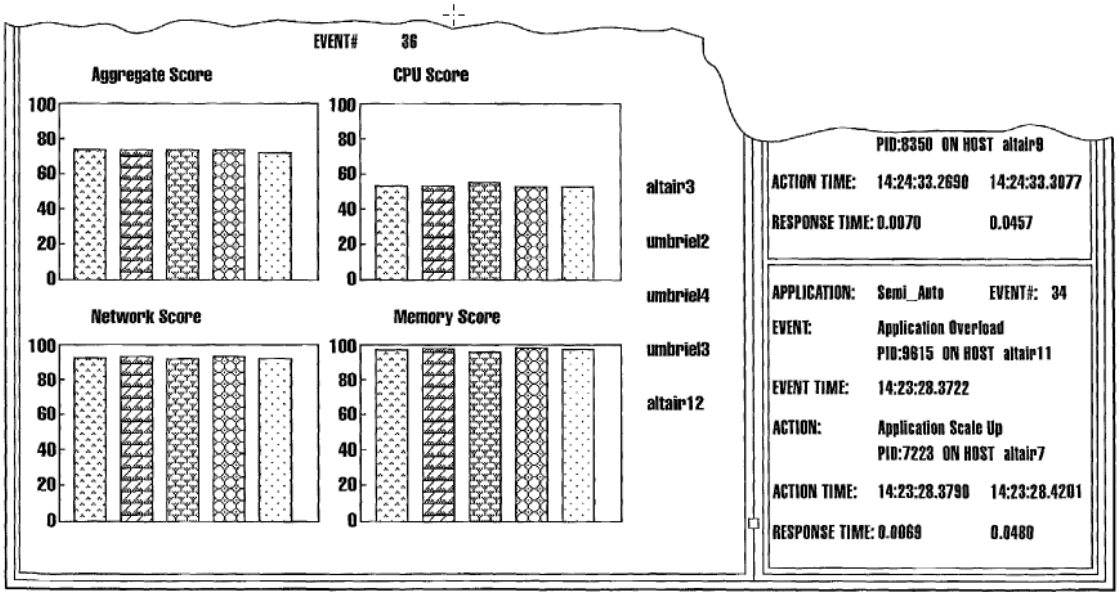


FIG.9B

Figures 9A and 9B represent a screen capture of Masters' "Decision Review Display," which "can provide a summary of allocation and reallocation actions taken by" the RMA. Ex. 1004, 30:1–4. "For each action, timing information regarding how long it took the Resource Management functions, e.g., the Resource Manager . . . and the Program

Controller . . . , to both arrive at a decision and to enact the decided action are shown along with host fitness scores that were used in arriving at the allocation decision." Ex. 1004, 30:4–10.

III. ANALYSIS

A. Legal Standards

The Supreme Court set out a framework for assessing obviousness under 35 U.S.C. § 103 that requires consideration of four factors: (1) the "level of ordinary skill in the pertinent art," (2) the "scope and content of the prior art," (3) the "differences between the prior art and the claims at issue," and (4) "secondary considerations" of non-obviousness such as "commercial success, long felt but unsolved needs, failure of others, etc."[6] *Graham v. John Deere Co. of Kan. City*, 383 U.S. 1, 17–18 (1966).

B. Level of Ordinary Skill in the Art

In determining the level of skill in the art, we consider the type of problems encountered in the art, the prior art solutions to those problems, the rapidity with which innovations are made, the sophistication of the technology, and the educational level of active workers in the field. *Custom Accessories, Inc. v. Jeffrey-Allan Indus., Inc.*, 807 F.2d 955, 962 (Fed. Cir. 1986); *Orthopedic Equip. Co. v. United States*, 702 F.2d 1005, 1011 (Fed. Cir. 1983).

Petitioner contends a person of ordinary skill in the art ("POSITA") at the time of the invention would have been a person with "a bachelor's degree in electrical engineering, computer science, or a similar field with at least two years of experience in distributed computing systems or a person

[6] The record does not include arguments or evidence regarding secondary considerations.

with a master's degree in electrical engineering, computer science, or a similar field with a specialization in distributed computing systems." Pet. 25 (citing Ex. 1003 ¶¶ 46–47). Petitioner further states that a person of ordinary skill in 2004 would have had

> the knowledge and skills necessary to: (i) create a distributed computing system with a network of computing devices/servers; (ii) implement and use software for the monitoring of distributed applications; (iii) understand how to define the events that are monitored as part of the performance monitoring system; (iv) implement a monitoring system that allows for the storage of the monitoring data being created; and (v) create reports that correlate performance information and data from the monitoring system to allow the user to assess and visually review the performance of the system.

Pet. 25–26 (citing Ex. 1003 ¶ 48).

Patent Owner contends that a person of ordinary skill in the art would have had "a bachelor's degree in Electrical Engineering, Computer Science, or a similar discipline, with one or two years of experience in this or a related field" and "would have been familiar with computing environments and distributed computing systems." PO Resp. 22–23 (citing Ex. 2001 ¶ 24). Patent Owner states that Petitioner's description of a POSITA is "similar" but that it "includes some additional requirements that Patent Owner does not deem necessary." PO Resp. 23. Further, Patent Owner states that any differences between the parties' view of a POSITA are not material. PO Resp. 23 (citing Ex. 2001 ¶ 25).

We determine a POSITA would have had "a bachelor's degree in Electrical Engineering, Computer Science, or a similar discipline, with one or two years of experience," which is consistent with the parties' arguments and testimony of the parties' declarants, as well as the level reflected in the '722 patent and the asserted prior art. We determine it is unnecessary to

reach the parties' additional statements of the knowledge and skills that a POSITA would allegedly possess.

A. Claim Construction

1. "Performance Data"

The parties did not expressly address the scope and meaning of "performance data" prior to our Final Written Decision. The Federal Circuit's Decision provides guidance regarding the scope and meaning of "performance data," and the parties address the issue in subsequent briefing. First, citing dependent claims 2, 3, and 4, the Federal Circuit stated that "[t]he recitation of applications and application tiers in the context of performance data in these dependent claims is persuasive evidence that the 'performance data' of claim 1 is not limited to instance-level data, even though the performance data must be '*associated with*' the performance of the one or more application instances." *Netflix*, 2024 WL 3507601 at *5. The Federal Circuit additionally cites the '722 patent's specification as evidence that "performance data" is not limited to instance-level data and "can include application-level and/or application-tier-level data. *Id.* (citing Ex. 1001, 1:56–2:17). Also, the Federal Circuit stated that "'performance data' includes data pertaining to the performance of a server and/or a technology tier." *Id.* (citing Ex. 1001, 3:56–57, 4:3–8, 4:14–18, 7:26–29).

In its briefing on remand, Petitioner contends "performance data" means "data relating to the performance of one or more application instances, including instance-level, application-level, device-level, and tier-level data." Pet. Remand Br. 2. Petitioner states that its proposed construction encompasses CPU utilization data (*id.*) and "timing metrics that measure how long it takes a computer system to perform a task—e.g.,

processing time, response time, and latency" (*id.* at 3 (citing Ex. 1001, 1:22–25, 3:62–67; Ex. 1028, 83:20–84:25)). Further, according to Petitioner, "'[p]erformance data' thus encompasses, for example, 'how long an application instance . . . takes to be created, to migrate . . . and to be destroyed." *Id.* at 3 (quoting Ex. 1001, 7:38–41; citing Ex. 1028, 75:24–76:16).

Patent Owner states on remand that the Board should interpret "performance data" as "data, including application instance-level, application-level, device-level, and tier-level data." PO Remand Resp. 2. Patent Owner argues "Petitioner's proposed construction is identical to [Patent Owner's] proposal except that it adds the unnecessary phrase 'relating to the performance of one or more application instances,'" which "would be superfluous and risk improperly broadening claim scope." *Id.* Patent Owner additionally argues that Petitioner's arguments that performance data "encompasses CPU utilization data" and "timing metrics that measure how long it takes a computer system to perform a task" are irrelevant. *Id.*

In its Remand Reply, Petitioner contends Patent Owner's construction would improperly remove "performance" from the definition. Pet. Remand Reply 1. In its Remand Sur-reply, Patent Owner argues Petitioner's definition would improperly broaden the claim language because "related to" is different than "associated with" and would eliminate the antecedent basis established by the word "said." PO Remand Sur-reply 1.

We find Petitioner's proposed construction to be generally consistent with the intrinsic record (including the claim language) and the Federal Circuit's guidance. However, we decline to adopt the use of "related to"

instead of "associated with" in Petitioner's proposed construction. Consistent with the Federal Circuit's guidance, we determine the proper construction of "performance data" further includes data pertaining to the performance of a server and/or a technology tier. *Netflix*, 2024 WL 3507601 at *6 (citing Ex. 1001, 3:56–57, 4:3–8, 4:14–18, 7:26–29). Accordingly, we determine "performance data" means "data associated with the performance of one or more application instances, including instance-level, application-level, device-level, and tier-level data and includes data pertaining to the performance of a server and/or a technology tier."

2. "Migrate/Migrating"

Patent Owner argues "a migration event requires the **same** application instance to be migrated (moved) from one location to another location." PO Resp. 11. According to Patent Owner, the '722 patent describes an application instance migrating from one server to another server, citing an example in which "application instance 320A may cease operation on server 300B and may be resumed **with the same data and configuration (*e.g.*, state)** on server 300A." *Id.* at 7 (quoting Ex. 1001, 6:10–13) (emphasis added by Patent Owner). Further, citing an "exemplary embodiment for a migration event," Patent Owner contends "it is clear from this disclosure that in each such embodiment the **same** application instance is migrated (moved) from one location to another based on certain unique identifiers of the specific application instance – whether certain instance configuration, instance data, and/or an instance ID." *Id.* at 8–9 (citing Ex. 1001, 6:49–7:2). Patent Owner argues the specification further makes clear that the exemplary application instance of Figure 6 "possesses uniquely defining features that allow the performance monitoring system to track this

specific application instance." *Id.* at 10 (citing Ex. 1001, 6:33–48, Fig. 6; Ex. 2001 ¶ 132).

Petitioner contends "'[m]igrating' an application instance means moving an instance from one location to another." Reply 7. Petitioner argues Patent Owner's arguments seek to impose additional requirements on "migration" that would require (i) moving an application instance by ceasing execution on the original server before starting the instance on the new server and (ii) "uniquely defining features" about the instance that must be transferred from the original host to the new host to verify that the instance remains the "same." Reply 8 (citing PO Resp. 8–10, 59–64). Petitioner argues that the '722 patent provides that migration can be performed in various ways and that the patent does not exclude any method of migration. *Id.* at 9.

We agree with Petitioner that Patent Owner's proposed definition would improperly read limiting language from the specification into the claims. First, claims 6 and 14 recite "wherein the migration event comprises the application instance migrating from a first location of execution to a second location of execution," without reciting or otherwise specifically requiring that it is the "same" instance or that the second application instance includes the same data or configuration. Further, because claims 8 and 16 narrow independent claims 6 and 14 by expressly reciting that "wherein said detecting a migration event comprises tracking the application instance with an application ID," it follows that the recited migration events of independent claims 6 and 14 do not require tracking the application instance with an application ID.

Moreover, after discussing different examples and embodiments of application instances with and without specific identifying features, the '722 patent expressly states that "application instance 120 in FIG. 6 is purely exemplary, and that in various embodiments, an application instance 120 may not have an instance ID 600, instance configuration 610, or instance data 620." Ex. 1001, 7:3–6; *see id*. at 6:33–35 (noting that an application instance "may" comprise an instance ID, an instance configuration, and instance data"), 6:52–55 (describing one example in which "performance monitoring system 200 may track an application instance migrating from one server to another" without reference to any instance identifier), 6:55–61 ("one embodiment" in which "performance monitoring system 200 may be operable to track a migrating application instance 120 by an associated instance ID 600"), 6:66–7:2 ("Alternatively, it is noted that performance monitoring system 200 may be operable to detect the migration of an application instance based on instance configuration 610 or instance data 620.").

Accordingly, based on the complete record, we adopt Petitioner's proposed construction that "'[m]igrating' an application instance means moving an instance from one location to another," which is consistent with the claim language itself.

3. "Application Instance"

Petitioner contends "application instance" is defined by the claim language to mean "a computer program executing on a computer system." Pet. 26. Patent Owner states that "application instance" does not require construction and that Petitioner's proposed construction is consistent with the claim language itself and the term's plain and ordinary meaning. PO

Resp. 6. We note that the Federal Circuit stated in its general discussion of the '722 patent that "[a]n application is a piece of software. If a piece of software is installed and run on a computer, the software running on the computer is an 'application instance.'" *Netflix*, 2024 WL 3507601 at *1, n.1 (citing Ex. 1001, 1:8–49, 10:2–3).

With that backdrop, we agree with the parties that the claim language itself sufficiently defines the term "application instance" and that it is unnecessary to further define the term.

4. *"Instance ID"*

The parties agree that "instance ID" as used in dependent claims 8 and 16 means "a piece of data, such as a number or alphanumeric string, operable to uniquely identify a specific application instance." PO Resp. 11–12 (citing Ex. 1001, 6:35–37; Ex. 2001 ¶ 136; Ex. 2004, 53:17–54:11); Reply 9.

B. Dependent Claims 4, 12, and 20

Claim 4 depends from claim 2, which depends from claim 1; claim 12 depends from claims 10, which depends from claim 9; and claim 20 depends from 18, which depends from claim 17. The Petition includes a limitation-by-limitation comparison of Masters to each limitation of claims 4, 12, and 20, including each base claim, with citations to the Declaration of Prashant Shenoy (Ex. 1003) as additional support. Pet. 28–72.

Patent Owner argues Masters does not disclose the "Correlating Limitations" (e.g., Limitation 1[d]) or the "Storing Limitations" (e.g., Limitation 1[e]) of claims 1, 9, and 17. PO Resp. 19–49. Patent Owner also argues Masters does not disclose the limitations of dependent claims 2, 10,

and 18 (PO Resp. at 50–53), nor the limitations of dependent claims 4, 12, and 20 (PO Resp. at 53–55).

1. Base Claims 1, 9, and 17 – Preambles/Preliminary Elements

1[a]. A method, comprising:
9[a]. A computer accessible medium, including program instructions executable to implement a method comprising:
17[a]. A performance monitoring system, comprising: a processor; a memory coupled to the processor for storing program instructions executable by said processor to implement:

The parties do not specifically address whether the preambles of claims 1, 9, and 17 are limiting. Petitioner states that to the extent the preambles are limiting, Masters teaches each preamble with its "system that monitors the performance of 'hosts, networks, and applications within a distributed computing environment.'" Pet. 29 (quoting Ex. 1004, 4:8–12). Petitioner additionally cites Masters as teaching a system embodied in software stored in memory and executed by a processor. Pet. 28–33 (citing, e.g., Ex. 1004, 1:31–34, 55:41–56:33, Figs. 1A, 1B, 15; Ex. 1003 ¶¶ 63–65).

Patent Owner does not address or rebut Petitioner's showing for the preambles and preliminary elements. We determine Petitioner shows Masters teaches the preamble limitations, to the extent they are limiting.

2. Collecting Limitations

> [1b] collecting performance data for one or more application instances, wherein the performance data is associated with the performance of said one or more application instances, wherein each application instance is a computer program executing on a computer system;
>
> [9b] collecting performance data for one or more application instances, wherein the performance data is associated with the performance of said one or more application instances, wherein each application instance is a computer program executing on a computer system;
>
> [17b] collecting performance data for one or more application instances, wherein the performance data is associated with the performance of said one or more application instances, wherein each application instance is a computer program executing on a computer system;

Petitioner contends Masters teaches the collecting limitations with its descriptions of collecting performance data "for all copies of a scalable application—i.e., for each 'application instance.'" Pet. 33–34 (citing Ex. 1004, 2:44–52, 4:8–12, 5:48–51, 6:64–7:2; Ex. 1003 ¶¶ 73–74). Petitioner states Masters teaches that application status and performance data is collected from each application instance and transmitted to the Resource Management hardware. Pet. 35–36 (citing Ex. 1004, Figs. 1A, 1B). Petitioner further contends that Masters teaches that different "functional groups" in the Resource Management Architecture are part of the collection of performance data. Pet. 36–37 (citing Ex. 1004, 7:33–58, 9:28–42, 10:18–31, 12:1–10; Ex. 1003 ¶¶ 78, 88).

Among other cited portions, Petitioner cites Masters' teachings of "Application Level Instrumentation" and an "instrumentation function group provides general-purpose application event reporting and event correlation

capabilities. Capabilities are provided for collecting and correlating application-provided data such as application statuses, states, performance, and internally detected errors." Pet. 36–38 (citing Ex. 1004, 7:19–20, 7:44–58, 12:1–10, Figs. 2A, 2B).

Petitioner also states that Masters teaches that performance data is associated with a "process id" for tracking specific application instances. Pet. 38 (citing Ex. 1004, 22:63–23:13; Ex. 1003 ¶ 82). Further, Petitioner contends Masters' "Quality of Service" managers use the performance data to determine whether copies of an application should be made, terminated, or moved. Pet. 38 (citing Ex. 1004, 18:43–48, 18:56–58, 22:63–23:27, 39:58–40:25; Ex. 1003 ¶ 114). And Petitioner cites examples of performance data that is collected, including "inter-processing times, throughputs, and latencies," as well as "host computer performance data" and "network performance data" such as "CPU utilization, memory utilization, network packets in, network packets out, and paging activity." Pet. 38 (citing Ex. 1004, 7:33–43, 8:16–34, 9:28–42,10:18–31, 18:24–39, 29:51–56, 45:36–48, Fig. 9B; Ex. 1003 ¶¶ 85–88).

Patent Owner does not directly rebut Petitioner's showings for the Collecting Limitations, but Patent Owner contends Petitioner is not consistent with its mapping of the Collecting Limitations and the Correlating Limitations. *See, e.g.*, PO Resp. 30–31, 38–39. We determine Petitioner shows Masters teaches the collecting limitations. We address Patent Owner's arguments about the consistency of Petitioner's mapping below with our discussion of the Correlating Limitations.

3. Detecting Limitations

[1c] detecting one or more instance life cycle events associated with said one or more application instances, wherein said one or more

> instance life cycle events comprise at least one of: the creation of at least one of said one or more application instances, the destruction of at least one of said one or more application instances, and the migration of at least one of said application instances;
>
> [9c] detecting one or more instance life cycle events associated with said one or more application instances, wherein said one or more instance life cycle events comprise at least one of: the creation of at least one of said one or more application instances, the destruction of at least one of said one or more application instances, and the migration of at least one of said one or more application instances;
>
> [17c] detecting one or more instance life cycle events associated with said one or more application instances, wherein said one or more instance life cycle events comprise at least one of: the creation of at least one of said one or more application instances, the destruction of at least one of said one or more application instances, and the migration of at least one of said one or more application instances;

Petitioner contends Masters teaches the detecting limitations with its "Application Control" function that detects when an application instance starts and stops as a result of decisions to "scale up" and "scale down" instances of the application. Pet. 40. Among other things, Petitioner asserts that a person of ordinary skill would have understood that "starting/scaling-up copies of an application is a 'creation' event and stopping/scaling-down copies of an application is a 'destruction' event." Pet. 40–41 (citing Ex. 1004, 33:56–63, 53:7–14; Ex. 1003 ¶ 89). Petitioner states that Masters' "Application Control functional group [FG5]" controls starting, stopping, and configuring applications on each host in the distributed computing environment and "Program Control FG50 of FG5" sends out immediate notifications when applications are started up, are shutdown, or fail. Pet. 41–42 (citing Ex. 1004, 26:39–43). Petitioner also contends Masters teaches detecting life cycle events and sending "application status updates"

upon the detection of faults and critical errors. Pet. 42–44 (citing Ex. 1004, 8:53–56, 24:5–6, 24:26–29, 24:42–44, 26:15–20, 32:65–33:8; Ex. 1003 ¶ 92).

Patent Owner does not expressly address or rebut Petitioner's showings regarding the detecting limitation, but, as explained in our analysis of the correlating limitation below, Patent Owner's arguments regarding the correlating limitation reflect disagreements with Petitioner's position that Masters teaches life cycle events, as claimed. We determine Petitioner shows Masters teaches the detecting limitations. We address Patent Owner's arguments about life cycle events below with our discussion of the Correlating Limitations.

4. Correlating Limitations

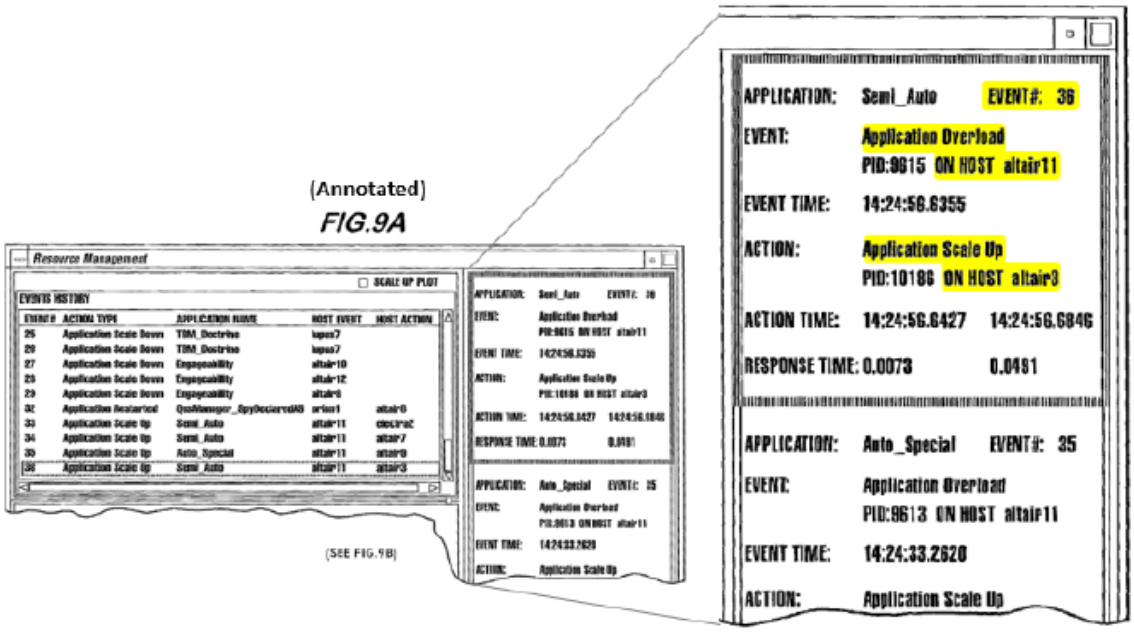
> [1d] correlating said performance data to said one or more instance life cycle events; and
>
> [9d] correlating said performance data to said one or more instance life cycle events; and
>
> [17d] correlating said performance data to said one or more instance life cycle events; and

Petitioner contends that the correlating limitations would have been obvious in view of Masters because, according to Petitioner, "Masters teaches that the RMA correlates performance data with the starting/scaling-up (i.e., creation) and stopping/scaling-down (i.e., destruction) of application instances." Pet. 46 (citing Ex. 1003 ¶ 94). Petitioner contends that Masters specifically teaches that its RMA correlates application statuses, states, performance, and internally detected errors. *Id.* (citing Ex. 1004, 7:48–49, 12:7–10). According to Petitioner, "[t]he 'application statuses' that are correlated with performance data include indications 'of any applications

which have been started or stopped.'" Pet. 46–47 (citing Ex. 1004 22:48–53, 39:42–47, 33:56–63, 53:7–14). Petitioner contends Masters teaches "Instrumentation Correlators" that receive application data and correlate the performance data with different events into higher-level metrics for use by displays or other functions of the RMA. *Id.* (citing Ex. 1004, 7:55–58, 12:15–19, 13:18–24; Ex. 1003 ¶ 95).

Petitioner also contends that Masters use of correlated data for reports and displays, such as the "'Resource Management Decision Review Display[s],' provide users with performance data correlated to 'allocation decisions'—i.e., the decision to start, stop, or move application instances." Pet. 48–49 (citing Ex. 1004, 18:24–47, 20:37–49, 29:36–40, 30:1–10, Figs. 9A, 9B; Ex. 1003 ¶ 97–99). According to Petitioner, Masters' Figure 9A, copied below with Petitioner's highlighting, "provides an example of a "Resource Management Decision Review Display" showing performance data.

(Annotated)
FIG.9A



Pet. 48–49. Masters' Figure 9A represents "a screen capture of the Resource Management Decision Review Display, which provides a summary of allocation and reallocation actions taken by the Resource Manager." Ex. 1001, 3:22–25.

Petitioner contends that "Figure 9A depicts 'Event #36' (highlighted in yellow), which is associated with the creation of a new application instance ('Application Scale Up') on host 'altair3.'" Pet. 49. According to Petitioner, Figure 9A indicates that the life cycle event was triggered for load balancing reasons, as indicated by the "Event: Application Overload" on host "altair11." Pet. 48. Further, Petitioner states that Figure 9A "shows that Event 36 correlates the 'Application Scale Up' life cycle event (i.e., the creation of a new application instance) with performance data, such as the response time for creating the new application instance and the time-of-day of the creation event." Pet. 49 (citing Ex. 1004, 20:37–44, 34:60–35:3, Fig. 9A; Ex. 1003 ¶¶ 102–103).

Petitioner also cites Masters' Figure 9B, copied below with Petitioner's annotations.

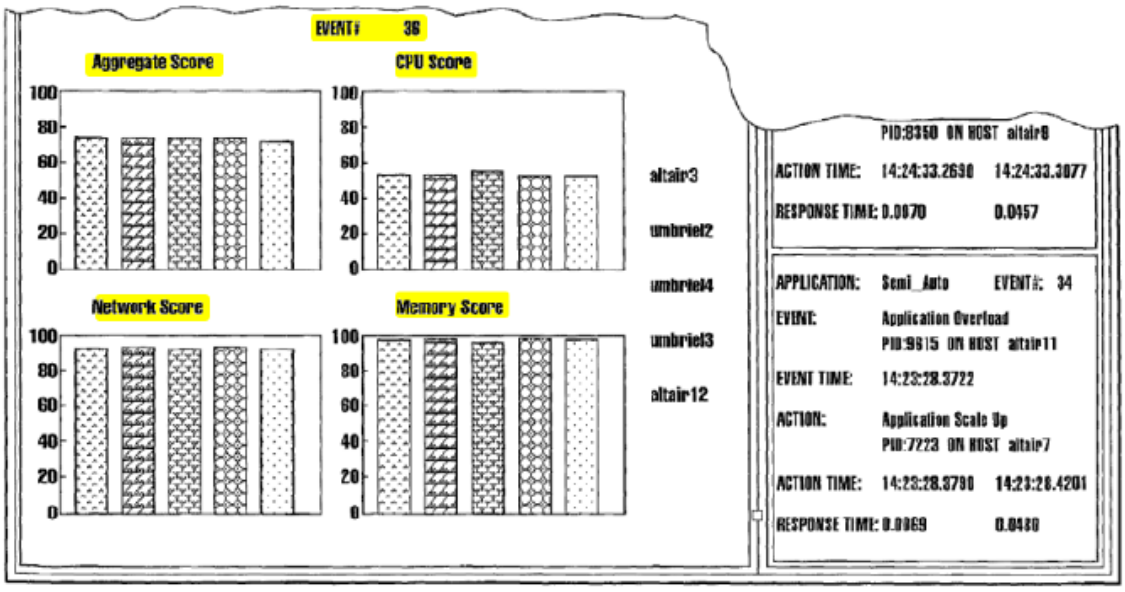
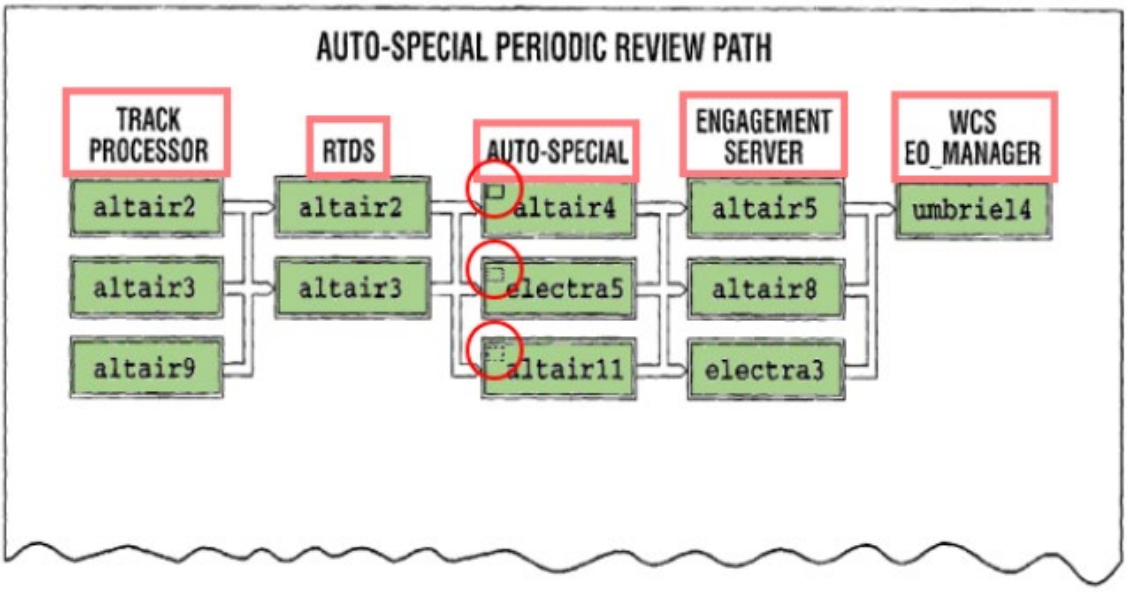


FIG.9B

Pet. 50. Masters' Figure 9B represents "a screen capture of the Resource Management Decision Review Display, which provides a summary of allocation and reallocation actions taken by the Resource Manager." Ex. 1001, 3:22–25. According to Petitioner, Figure 9B is another example of "performance data correlated with the 'Application Scale Up' event (Event 36), including the 'aggregate' health score, and the memory, CPU, and network health for various hosts in the distributed computing system at the time of the life cycle event." Pet. 50 (citing Ex. 1003 ¶ 104).

Petitioner additionally cites the "Path Displays" of Figures 8A and 8B. Pet. 50–55. Copied below are Petitioner's annotated versions of Figures 8A and 8B.

FIG.8A
(Annotated)



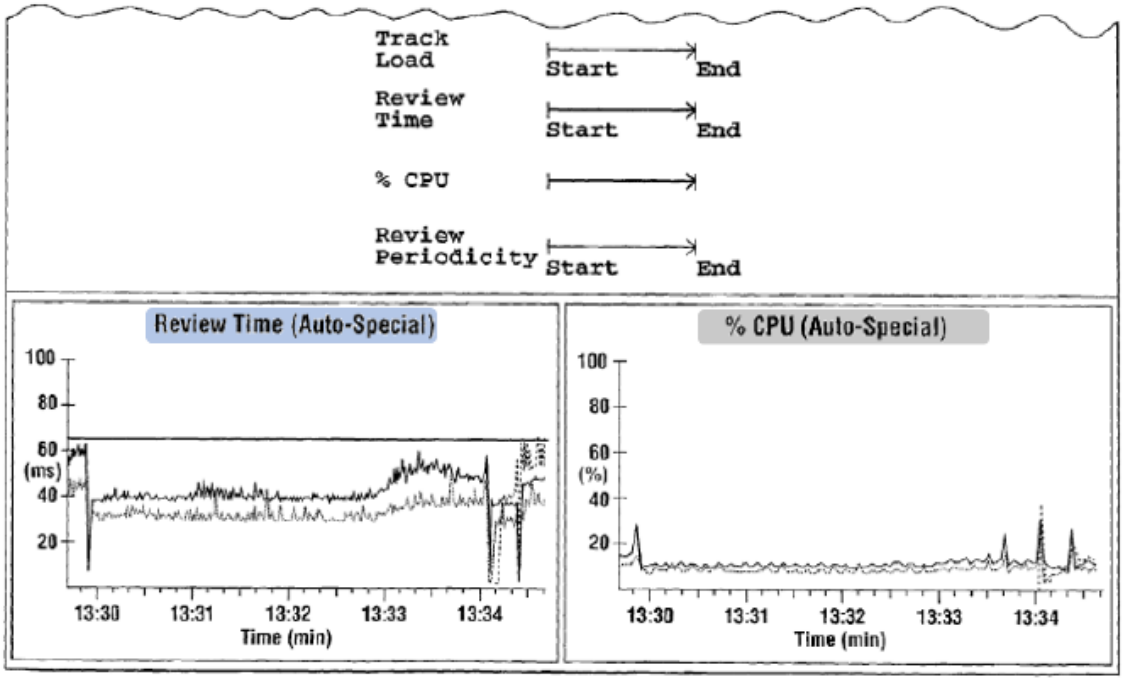


FIG.8B
(Annotated)

Pet. 53–54. Masters' Figures 8A and 8B "represent a screen capture of a path display generated by the Resource Management Architecture." Ex. 1001, 3:19–21.

According to Petitioner, Figure 8A depicts five applications (boxed in red) and 12 application instances (highlighted in green) (Pet. 51–52), and Petitioner states that "three instances of the 'Auto-Special' application have been selected for reporting, as indicated by the broken-line rectangles shown for those instances (circled in red)" (Pet. 52). Petitioner states that Figure 8B "correlates an application instance life cycle event—the creation of a new application instance on altair11 at approximately 13:34 [Time (min)]—with the performance of each of the application instances executing the 'Auto-Special' application." Pet. 54 (citing Ex. 1003 ¶ 111). Petitioner contends Figure 8B correlates the creation of that application instance shortly after 13:34 with the performance of each of the "Auto-Special" instances, and "upon creation of the third application instance, the review time of the two other application instances drops (at first dramatically), while the review time for the new instance climbs." Pet. 53–54. Petitioner additionally argues that to the extent Masters does not directly teach correlating, "a POSITA would have been motivated to correlate these life cycle events with performance data to verify that creating/destroying/migrating an application instance solved the performance issue that triggered the life cycle action." Pet. 58–59 (citing Ex. 1003 ¶ 120).

Patent Owner argues Masters does not teach or suggest the Correlating Limitations and that Petitioner's mapping of the claim elements to the Correlating Limitations is not consistent with the antecedent bases of

each claim element. PO Resp. 20–50. According to Patent Owner, "the timing information shown in FIGS. 9A and 9B indicate (a) how long the RMA (through FG42) took to make a decision and (b) how long the RMA (through FG50) took to implement the decision." PO Resp. at 23 (citing Ex. 1004, 30:1–19; Ex. 2001 ¶¶ 63–65). More specifically, Patent Owner states that in Figure 9A, "RESPONSE TIME 0.0073 is the amount of time it took the RMA (through FG42) to make a decision to start up an application and RESPONSE TIME 0.0491 is the amount of time it took the RMA (through FG50) to implement this decision—*i.e.*, these numbers are response times of the RMA, not the response times of any application instance." PO Resp. 27 (citing Ex. 2001 ¶¶ 70–74); *see* Sur-reply 6–8; PO Remand Resp. 4. Patent Owner similarly argues Figure 9B shows "'host-level performance data' for various hosts (*i.e.*, machines) – not performance data for application instances." PO Resp. 29. Further, Patent Owner argues "the data displayed on FIGS. 9A and 9B cannot satisfy the Collecting Limitations of the Challenged Claims" and "it is impossible for FIGS. 9A and 9B to satisfy the Correlating Limitations because to do so would require FIGS. 9A and 9B to show a correlation between data associated with the performance of the scaled up application from Event #36 and the scaled up event itself – **performance data that is simply not contained in FIGS. 9A and 9B**." PO. Resp. 32; *see* Sur-reply 7–8. Patent Owner additionally argues that nothing in Figures 9A and 9B shows any type of relationship established between the scaled up event of Event # 36 and any other data whatsoever." PO Resp. 39 (citing Ex. 2001 ¶ 80); PO Remand Resp. 4–5. According to Patent Owner, the performance data and the life cycle event must *both* be associated with *the same* application instance, but Patent Owner argues

Masters' CPU data and response time have "no connection to the performance of a new application instance because the new application instance does not perform until after the response time has passed." PO Remand Sur-reply 2–4.

Addressing Figures 8A and 8B, Patent Owner argues "it is entirely unclear whether the data shown in the Review Time (Auto-Special) and/or % CPU (Auto-Special) graphs are performance data associated with the Auto-Special application" and "Petitioner has not satisfied its burden to demonstrate that FIG. 8B shows the collection of performance data for an application instance." PO Resp. 39 (citing Ex. 2001 ¶¶ 84, 88–89); Sur-reply 10–12. Patent Owner also argues that Petitioner has not carried its burden to show that "Figure 8B depicts a detection of a life cycle event associated with the same one or more application instances for which the system collects performance data." PO Resp. 39. According to Patent Owner, "FIG. 8B shows certain performance data for the copies of application Auto-Special on the three hosts from the period just before 13:30 to just after 13:34" and "[w]hile these graphs indicate that an application instance on host altair11 begins functioning at approximately 13:34, nothing in FIGS. 8A or 8b indicate[s] that this application instance was 'created' at 13:34." PO Resp. 40; *see* PO Remand Resp. 6–7, 10. Rather, quoting Dr. Rosenblum, Patent Owner argues that Figure 8B depicts that host altair11 "began to service some sort of load at around 13:34" and "[i]t is equally likely, if not more likely, that the copy of Auto-Special on altair11 was in existence during the entire period depicted on Fig. 9B, and the RMA merely shifted load to the copy of the Auto-Special application on altair11 in response to the increased overall load on the system." PO Resp. 40–41

(quoting Ex. 2001 ¶¶ 87, 91); *see* PO Remand Sur-reply 5 (arguing that the Board previously held that Figure 8B does not indicate that a life cycle event occurred at 13:34). Further, Patent Owner argues Petitioner's declarant Dr. Shenoy was unable to explain what information is depicted in Figure 8B. PO Resp. 41–44; PO Remand Resp. 6–7.

Contrary to Patent Owner's argument, Masters depicts the creation of an application instance and related performance data in Figures 8A and 8B. This is confirmed by Masters' Appendix J[7] (Ex. 1024), which includes the screen capture copied below:

[7] Appendix J was filed on a CD-ROM with the patent application that ultimately issued as Masters (US Patent No. 7,051,098) and is part and parcel to Masters itself. *Netflix*, 2024 WL 3507601 at *8.

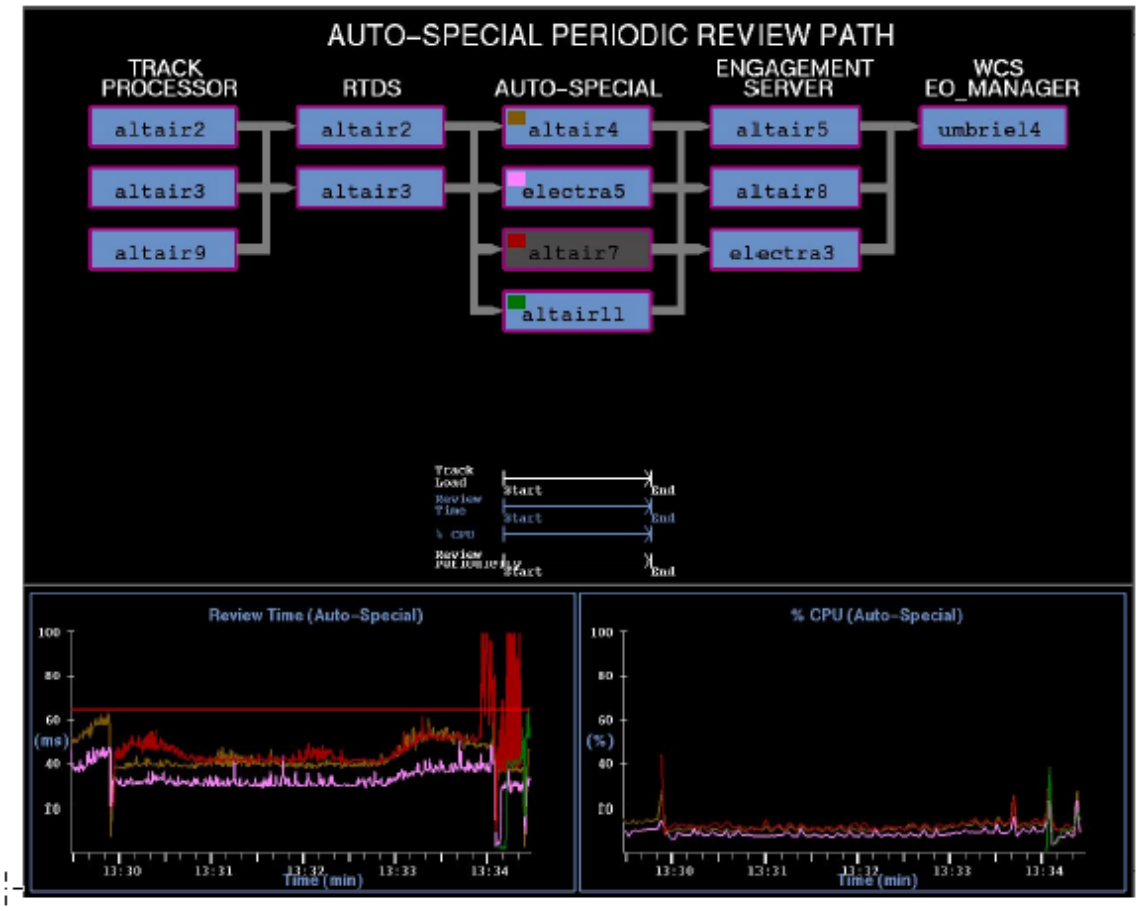


Figure 5.6.2.1-2

Ex. 1024, 38. Figure 5.6.2.1-2 in Masters' Appendix J "is a screen capture of the Path Display that shows a[] Resource Manager action was taken." Ex. 1024, 38. The Figure "shows the review times in the first graph (from the left) and the CPU utilization by each process in the second graph, corresponding to the blue lines in the center of the screen." Ex. 1024, 39. "The color of the lines in the graph correspond to the colors in the corner of the process boxes at the top of the screen in this case the Auto-Special processes." *Id.* The Figure shows that "[t]he Altair7 Auto-Special process clearly has longer review times than other processes, so the Resource Manager terminates this copy and starts another on Altair11. The green line at the edge of the first graph indicates the new process' review time." *Id.*

Applying our construction of "performance data," we agree with Petitioner that, among other things, the CPU utilization and review time shown in Masters' Figure 5.6.2.1-2 and Figures 8A and 8B fall within the scope of the claimed "performance data." *See* Pet. 54–56. The CPU utilization data and review time data are both data associated with the performance of one or more application instances. Further, the Figures and corresponding description in Appendix J evidence collection of the performance data for each application instance. Ex. 1024, 39. The startup of a copy of the Auto-Special process on Altair11 constitutes a life cycle event, as it is the creation of an application instance; in fact, the termination of Auto-Special on Altair7 is another example of a life cycle event, and together these termination and creation events constitute a migration event. *Id*. We determine that these Figures and related descriptions in Masters teach collecting performance data and correlating the performance data to the life cycle events.

We also agree with Petitioner that Masters' Figures 9A and 9B similarly show the collection and correlation of performance data, at least with the response time, action time, CPU score, and similar metrics depicted in the Figures. *See* Pet. 47–50. Figures 9A and 9B show the correlation of the performance data to a life cycle event, by, for example, depicting the performance issue (an overload on Altair11) that cause the creation of a new application instance on Altair3. We determine that the response time to trigger the creation of an application instance on Altrair3 and the amount of time the application instance takes to start are two examples of performance data correlated with the life cycle event. *See Netflix*, 2024 WL 3507601 at *2, n.5. As stated in Masters, Figures 9A and 9B depict Masters' Decision

Review Display that provides a summary of allocation and reallocation actions taken by the RMA, and "[f]or each action, timing information regarding how long it took the Resource Management functions, e.g., the Resource Manager . . . and the Program Controller . . ., to both arrive at a decision and to enact the decided action are shown along with host fitness scores that were used in arriving at the allocation decision." Ex. 1004, 30:1–10. We disagree with Patent Owner's arguments that the Masters' "response time" "cannot be associated with the performance of the 'Semi-Auto' application instance because the response time passes before the instance exists" (PO Remand Br. 4), and we agree with Petitioner that the time required to start an application instance is a metric associated with the creation of that instance.

In sum, we agree with Petitioner that Masters teaches the Correlating Limitations.

5. Storing Limitations

[1e] storing the correlated performance data.
[9e] storing the correlated performance data.
[17e] storing the correlated performance data.

Among other things, Petitioner contends Masters teaches the use of history servers to store performance data. Pet. 61 (citing Ex. 1004, 5:67–6:36, 9:66–10:5, 19:53–59, 44:58–60, 46:41–44; Ex. 1003 ¶ 123). Petitioner also argues Masters teaches storing performance data correlated to life cycle events for use in displays and in making resource allocation recommendations. Pet. 62–64. Specifically, Petitioner contends that Figures 8A, 8B, 9A, and 9B display CPU usage, review time, response time, and other metrics correlated to the creation of a new application instance and

that a person of ordinary skill would have understood that such data is stored in a memory so that it can be displayed. Pet. 63.

Patent Owner argues that "the data set forth in FIGS. 8A–9B does not satisfy the Correlating Limitations" and "[t]herefore, Petitioner cannot rely on this data to satisfy the Storing Limitations." PO Resp. 46–47. Patent Owner additionally contends that Masters does not state that the information in Figures 8A, 8B, 9A, and 9C is stored in its History Servers, and Patent Owner argues displaying "real-time graphs" such as those in Masters' Figures "do[] not capture the correlation relationship to the extent there was any correlated relationship to begin with." *Id.* at 47–48 (citing Ex. 2001 ¶¶ 106–109). Patent Owner also argues that the Storing Limitations require "storing the performance data in some way that preserves the relationship established in the Correlating Limitations." Sur-reply 15–16.

We agree with Petitioner that Masters' Figures and related descriptions teach or suggest storing performance data correlated with the one or more life cycle events. Pet. 61–64. We find credible Dr. Shenoy's testimony that among other things, Masters teaches storing performance data correlated to life cycle events at least because Masters teaches displaying that the correlated performance data in displays and using the correlated performance data to make resource allocation decisions, and a POSITA would reasonably understand Masters' history servers to store such data. Ex. 1003 ¶¶ 123–127. Patent Owner and Dr. Rosenblum apply an incorrect construction of "performance data" to argue that storage of the data illustrated in Figures 8A, 8B, 9A, and 9B does not satisfy the Storing Limitations. We also agree with Petitioner that the Storing Limitations merely claim conventional activity that would have been obvious in view of

Masters' teachings. *See*

Pet. 63–64.

6. Conclusion Regarding Claims 1, 9, and 17

Having considered the complete record, for the reasons explained above, we determine Petitioner has shown that Masters teaches the limitations of claims 1, 9, and 17.

7. Base Claims 2, 10, 18

> 2. The method of claim 1, wherein the one or more application instances are instances of an application, and wherein said correlating said performance data to one or more instance life cycle events comprises correlating the one or more instance life cycle events to the performance of the application.
>
> 10. The computer accessible medium of claim 9, wherein the one or more application instances are instances of an application, and wherein said correlating said performance data to one or more instance life cycle events comprises correlating the one or more instance life cycle events to the performance of the application.
>
> 18. The performance monitoring system of claim 17, wherein the one or more application instances are instances of an application, and wherein said correlating said performance data to one or more instance life cycle events comprises correlating the one or more instance life cycle events to the performance of the application.

Petitioner argues Masters teaches that its RMA monitors the performance of instances of each application instance running on its host computers. Pet. 64–65. Petitioner further states that Masters teaches correlating the one or more instance life cycle events to the performance of the application for the same reasons discussed above with the Correlating Limitations of claims 1, 9, and 17. Pet. 66. As one example, Petitioner contends Figures 8A and 8B correlate the creation of a new application instance with the performance of the overall application Auto-Special. *Id.*

Patent Owner relies on the same arguments addressed above to assert that Masters' Figures 8A–9A fail to demonstrate any correlation that would satisfy the Correlating Limitations of claims 1, 9, and 17. PO Resp. 50–51. Patent Owner also argues that dependent claims 2, 10, and 18 require that the one or more life cycle events themselves are correlated to the performance of the application, and, according to Patent Owner, Masters only tracks performance data to determine whether to undertake a future scale up or scale down event. PO Resp. 52.

For the same reasons addressed above, we agree with Petitioner that Masters' Figures 8A and 8B teach correlating the termination of an application instance, creation of a new application instance, and migration of an application instance with the performance of the application, e.g., Auto-Special. We further agree with Petitioner that Masters teaches correlating one or more life cycle events to the performance of the application. Accordingly, we determine Petitioner has shown that Masters teaches the limitations of claims 2, 10, and 18.

8. Claims 4, 12, and 20

> 4. The method of claim 2, wherein said correlating said performance data to one or more instance life cycle events comprises determining the change in performance of the application as the one or more application instances migrate from one location to another location.
>
> 12. The computer accessible medium of claim 10, wherein said correlating said performance data to one or more instance life cycle events comprises determining the change in performance of the application as the one or more application instances migrate from one location to another location.
>
> 20. The performance monitoring system of claim 18, wherein said correlating said performance data to one or more instance life

> cycle events comprises determining the change in performance of the application as the one or more application instances migrate from one location to another location.

Petitioner contends Masters teaches that the RMA determines whether an application instance should be moved when it is necessary to improve performance. Pet. 67–68 (citing, e.g., Ex. 1004, 6:59–60, 7:44–49, 8:9–34, 15:63–66, 17:29, 18:4–9, 18:29–48, 19:4–12, 19:60–65, 20:60–62, 21:40–49, 22:28–32, 23:28–33, 33:63–34:9, 37:46–54, 39:8–19, 40:25–31, 53:14–26; Ex. 1003 ¶¶ 133–135). Petitioner also contends that Masters' Figures 8A and 8B and related descriptions teach migration as a scale-up request of a new application instance and a scale-down of a badly performing copy while continuously monitoring performance metrics to determine the change in performance during the migration event. Pet. 69–72 (citing Ex. 1004, 6:33–36, 19:20-27, 20:58–65, 23:28-33, Figs. 8A, 8B; Ex. 1003 ¶¶ 141–143).

In addition to its arguments addressed above regarding the Correlating Limitations of the case claims, Patent Owner relies on arguments presented with respect to claims 6, 14, and 22 to argue Masters does not disclose "migration." PO Resp. 53–55. We address those arguments below.

For the reasons explained above with respect to the base claims, and the additional reasons explained below with regard to Masters' teachings of migration, we determine Petitioner has shown by a preponderance of the evidence that claims 4, 12, and 20 are unpatentable in view of Masters.

C. Independent Claims 6, 14, and 22

1. Preambles/Preliminary Elements

> 6[a]. A method, comprising:

> 14[a]. A computer accessible medium, including program instructions executable to implement a method comprising:
>
> 22[a]. A performance monitoring system, comprising: a processor; a memory coupled to the processor and storing program instructions executable by said processor to:

The parties do not specifically address whether the preambles of claims 1, 9, and 17 are limiting. Petitioner states that to the extent the preambles are limiting, Masters teaches each preamble for the same reasons addressed above for Limitations 1[a], 9[a], and 17[a]. Pet. 73.

Patent Owner does not address the preambles/preliminary elements. We determine Petitioner shows Masters teaches the preamble limitations, to the extent they are limiting.

2. Monitoring Limitations

> 6[b]. monitoring performance of an application instance that is a computer program executing on a computer system, wherein said monitoring comprises collecting performance data associated with the performance of the application instance executing on the computer system;
>
> 14[b]. monitoring performance of an application instance that is a computer program executing on a computer system, wherein said monitoring comprises collecting performance data associated with the performance of the application instance executing on the computer system;
>
> 22[b]. monitor performance of an application instance that is a computer program executing on a computer system, wherein to monitor said performance the program instructions are executable to collect performance data associated with the performance of the application instance executing on the computer system;

Petitioner argues Masters teaches a monitoring system that collects data on the performance of all copies of a scalable application. Pet. 74

(citing Ex. 1004, 2:44–52, 16:22–25, 18:4–14). Further, Petitioner argues Masters teaches collecting performance data associated with the performance of application instances executing on a computer system for the same reasons discussed above for the Collecting Limitations (Limitations 1[b], 9[b], and 17[b]). Pet. 76–77.

Patent Owner does not directly address or rebut Petitioner's showings for the Monitoring Limitations. We determine Petitioner shows Masters teaches the Monitoring Limitations.

3. Detecting Limitations

> 6[c]. detecting a migration event associated with the application instance, wherein the migration event comprises the application instance migrating from a first location of execution to a second location of execution;
>
> 14[c]. detecting a migration event associated with the application instance, wherein the migration event comprises the application instance migrating from a first location of execution to a second location of execution;
>
> 22[c]. detect a migration event associated with the application instance, wherein the migration event comprises the application instance migrating from a first location of execution to a second location of execution;

Among other things, Petitioner argues Masters teaches detecting a life cycle event for the same reasons discussed above regarding the Detecting Limitations of claims 1, 9, and 17. Pet. 77. Petitioner also argues Masters teaches a migration event as a scale down of an application instance on a poorly performing host and a scale up of the application instance on a different host. Pet. 77–78 (citing Ex. 1004, 19:20–24, 19:42–59, 22:32–37, 22:48–53, 52:38–53:6). Further, Petitioner contends Masters Figure 9A depicts migrations of application instances from one host to another,

including restarting an application on a different host after a fault. Pet. 78–79 (citing Ex. 1004, 19:20–27, Fig. 9A; Ex. 1001, 6:13–16; Ex. 1003 ¶¶ 136, 160, 161).

First, Patent Owner argues Masters discloses moving an application, not an application instance. PO Resp. 57–58. Further, Patent Owner argues that even if "Masters'[s] concept of a 'copy' of an application is analogous to an application instance (which they are not), Masters never once discloses moving a discrete copy of the application from one location to another." PO Resp. 58 (citing Ex. 1004, 22:28–37; Ex. 2001 ¶ 154). Further, Patent Owner argues that because Masters discloses first performing a scale up (creating a second copy) at a new location and then a scale down (descrying the first copy) at the old location, it is not the same application instance that moves from one location to another because it is impossible for the same application instance to be in existence and executing at the same time. PO Resp. 59–60 (citing Ex. 1004, 22:32–37, 39:11–19; Ex. 2001 ¶¶ 155–157). Patent Owner additionally argues that Masters' process of restarting an application on a new host after a fault is not a migration because each copy of the application has its own unique process ID. PO Resp. 60–64 (citing Ex. 1004, Fig. 9A; Ex. 2004, 14:16–15:10, 15:20–26:11, 37:11–21, 38:6–18; Ex. 2001 ¶ 161).

As explained above, we determine that Masters' teaching of scaling down an application instance on one host and scaling up an application instance on another host constitutes a migration event as claimed, even if the scale up occurs immediately before the scale down. Among other things, like Masters, the '722 patent states that an application instance may be migrated due to a fail-over or for load-balancing (Ex. 1001, 6:5–32), and

nothing in the claims or the specification requires first scaling down the application instance before the scaling up on a different host. Further, Masters' teachings is not limited to scaling up before scaling down; in Appendix J, for example, Masters states "The Altair7 Auto-Special process clearly has longer review times than other processes, so the Resource Manager terminates this copy and starts another on Altair 11" (Ex. 1024, 39), and in Figure 8, for example, Masters depicts scaling up an application instance of Auto-Special on one host and scaling down an application instance of Auto-Special both within fractions of a second after 13:34 (Ex. 1004, Fig. 8B). Further, for the reasons explained above, we disagree with Patent Owner's arguments that would impose requirements of a "same" instance ID or similar identifier; the '722 patent's description of instance IDs, instance data, and configuration data is "purely exemplary." Ex. 1001, 6:49–7:13.

Accordingly, we determine Petitioner has sufficiently shown that Masters teaches the Detecting Limitation.

4. Continued Monitoring Limitations

6[d]. subsequent to said detecting, continuing to monitor the performance of the application instance;
14[d]. subsequent to said detecting, continuing to monitor the performance of the application instance;
22[d]. subsequent to detecting the migration event, continue to monitor the performance of the application instance;

Petitioner contends Masters teaches that its RMA continuously monitors the performance of application instances, as evidenced by Figures 8A and 8B, among other things. Pet. 80–82. Patent Owner relies on its

arguments for other limitations and does not specifically rebut Petitioner's showings for the Continued Monitoring Limitations. *See* PO Resp. 65.

We determine Petitioner has sufficiently shown that Masters teaches the Continued Monitoring Limitations.

5. Correlating Limitations & Storing Limitations

6[e]. correlating the performance data to the migration event; and;
14[e]. correlating the performance data to the migration event; and;
22[e]. correlate the performance data to the migration event; and;
6[f]. storing the correlated performance data.
14[f]. storing the correlated performance data.
22[f]. store the correlated performance data.

The parties rely on the same arguments addressed above for the Correlating Limitations and the Storing Limitations. Pet. 83–84; PO Resp. 65. For the same reasons explained above, we determine Petitioner has sufficiently shown that Masters teaches the Correlating Limitations and the Storing Limitations.

6. Conclusion Regarding Claims 6, 14, and 22

Having considered the complete record, for the reasons explained above, we determine Petitioner has proven by a preponderance of the evidence that claims 6, 14, and 22 are unpatentable as obvious in view of Masters.

D. Dependent Claims 7 and 15

> 7. The method of claim 6, wherein correlating performance data to the migration event comprises relating the performance data to the first and second locations.
>
> 15. The computer accessible medium of claim 14, wherein correlating performance data to the migration event comprises relating the performance data to the first and second locations.

Petitioner contends Masters teaches reporting performance data for host computers on which an application instance is terminated (a first location) and created (a second location) as part of a migration event. Pet. 84–89. Among other things, Petitioner cites Masters' Figures 8A, 8B, 9A, and 9B as evidencing relating the performance data to the first and second location. *Id.* (citing Ex, 1004, 8:9–30, 22:32–37, 24:47–50, 29:14–19, 20:1–10, Figs. 8A, 8B, 9A, 9B; Ex. 1003 ¶¶ 173, 175–184).

Patent Owner relies on its arguments for claims 6 and 14 and does not specifically address or rebut Petitioner's showings for claims 7 and 15. PO Resp. 67.

Based on the complete record, we determine Petitioner has shown by a preponderance of the evidence that claims 7 and 15 are unpatentable in view of Masters.

E. Dependent Claims 8 and 16

> 8. The method of claim 6, wherein said detecting a migration event comprises tracking the application instance with an instance ID.
>
> 16. The computer accessible medium of claim 14, wherein detecting a migration event comprises tracking the application instance with an instance ID.

Petitioner contends Masters teaches tracking a process ID and hostname in connection with the creation and termination of application

instances. Pet. 90 (citing Ex. 1003 ¶¶ 187–188). According to Petitioner, Masters teaches identifying each application instance with the hostname and process ID of the application, as shown in Masters' Figure 9A and related descriptions. Pet. 90–91 (citing Ex. 1004, 22:63–23:12, 39:58–40:10, Fig. 9A). Petitioner contends that while Figure 9A only shows the process ID for an example of a scale up event, "a POSITA would have understood that the same report would be provided for migration events." Pet. 91 (citing Ex. 1003 ¶ 191). Further, Petitioner contends that to the extent that Masters does not teach tracking an instance ID in connection with a migration event, "it would have been obvious based on Masters'[s] teachings to track the [process IDs] and hostnames assigned to application instances when those instances are moved to a new host." Pet. 92 (citing Ex. 1003 ¶¶ 192–194; Ex. 1004, 23:28–33)

Patent Owner argues Masters' process ID is unique to an individual host and that even a process ID combined with a hostname "would still not result in a situation where the same process ID and host name could be used to track the migrating application instance because the host name (as well as the process ID) would change when the application was restarted/scaled up at the new location under the teachings of Masters." PO Resp. 66–67. Further, Patent Owner argues that if a process ID does not remain constant during a migration event, it is not a piece of data uniquely identifying an instance of an application. Sur-reply 24.

As noted above, the parties agree that "instance ID" means "a piece of data, such as a number or alphanumeric string, operable to uniquely identify a specific application instance." In the context of a migration event, the '722 patent's specification states:

> In one embodiment, performance monitoring system 200 may be operable to track a migrating application instance 120 by an associated instance ID 600. For example, a migrating application instance 120 may be destroyed on a first server and recreated on a second server with the same instance ID 600, instance configuration 610 and instance data 620. Accordingly, performance monitoring system 200 may be operable to detect that a new application instance 120 has the same instance ID 600 as a previously destroyed application instance 120, and thereby determine that the application instance 120 has migrated from a first server to a second server.

Ex. 1001, 6:55–66. The example provided explicitly describes tracking an application instance migrating from one server to another server with application instance having the *same instance ID*, but that discussion is characterized as an "example" in "one embodiment." In contrast, rather than invoking language that mirrors the example by reciting, e.g., "the same instance ID," the claims' language is broader: "detecting a migration event comprises tracking the application instance *with an instance ID*." Thus, the claim language itself may be more broadly read to allow for the application instance to have different instance IDs during a migration event.

It is unnecessary to determine whether the claims' proper scope is limited to the specification's example, however, because we agree with Petitioner that under either interpretation, claims 8 and 16 would have been obvious to a POSITA in view of Masters' teachings of detecting and tracking life cycle events for application instances having data to uniquely identify the application instance such as process IDs and host names.

First, if the claims are broad enough to include tracking an application instance with an instance ID that can change from one server to another, Masters expressly teaches claims 8 and 16; Masters' Figures 9A and 9B, for example, depict a process ID and host name with each life cycle event.

Even if the application instance's process ID and host name changes from one location (host) to another location (host), Masters teaches detecting a migration event associated with the application instance, including tracking the application instance with an application instance ID from one location to another. *E.g.*, Ex. 1004, Figs. 9A, 9B.

If, as Patent Owner suggests, claims 8 and 16 should be read to require that tracking the application instance means, e.g., tracking an application instance's instance ID from one location to another to verify that it is the same instance ID, we agree with Petitioner that maintaining the same instance ID would have been obvious to a person of ordinary skill in the art. *See* Pet. 92. We credit Dr. Shenoy's testimony that, given Masters' teachings of an event number, process ID, and host name associated with an event, a POSITA would have been motivated to use such data to track application instances as they move from one host to another, to facilitate performance monitoring and to verify that the migration event resulted in the desired improvement of performance of a migrated application instance. Ex. 1003 ¶¶ 189–194. Dr. Rosenblum states that a POSITA would understand that Masters' process IDs are arbitrarily assigned by the operating system of the host and that it is unlikely that process IDs involved in an application instance's migration might be the same on different hosts. Ex. 2001 ¶ 167. Dr. Rosenblum does not, however, address Petitioner's proposed modification of Masters (*see id.*), and we are persuaded by Dr. Shenoy's testimony and agree with Petitioner that any modifications required to use the same data to track application instances from one location to another would have been minor and within the skill of a POSITA (Pet. 92; Ex. 1003 ¶¶ 189–194).

Accordingly, based on the entire record, we determine Petitioner has shown by a preponderance of the evidence that claims 8 and 16 are unpatentable as obvious in view of Masters.

IV. CONCLUSION

For the reasons explained above, we determine Petitioner has established by a preponderance of the evidence that claims 4, 6–8, 12, 14–16, 20, and 22 are unpatentable.

In summary:

Claims	35 U.S.C. §	Reference(s)/Basis	Claims Shown Unpatentable	Claims Not Shown Unpatentable
4, 6–8, 12, 14–16, 20, 22	103(a)	Masters	4, 6–8, 12, 14–16, 20, 22	
Overall Outcome			4, 6–8, 12, 14–16, 20, 22	

V. ORDER

In consideration of the foregoing, it is hereby:

ORDERED that claims 4, 6–8, 12, 14–16, 20, and 22 have been proven by a preponderance of the evidence to be unpatentable; and

FURTHER ORDERED that, because this is a final written decision, parties to this proceeding seeking judicial review of our Decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

FOR PETITIONER:

Harper Batts
Chris Ponder
Jeffrey Liang
SHEPPARD, MULLIN, RICHTER & HAMPTON LLP
HBatts@sheppardmullin.com,
CPonder@sheppardmullin.com
JLiang@sheppardmullin.com

FOR PATENT OWNER:

Daniel S. Young
QUARLES & BRADY LLP
dyoung@quarles.com

Chad E. King
KING IAM, LLC
chad@king-iam.com