

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

INTEL CORPORATION,
Petitioner,

v.

ADVANCED CLUSTER SYSTEMS, INC.,
Patent Owner.

IPR2025-00916
Patent 11,811,582 B2

Before KARL D. EASTHOM, HUNG H. BUI, and SCOTT RAEVSKY,
Administrative Patent Judges.

EASTHOM, *Administrative Patent Judge.*

JUDGMENT
Final Written Decision
Determining All Challenged Claims Unpatentable
35 U.S.C. § 318(a)

I. INTRODUCTION

Intel Corporation (“Petitioner”) filed a Petition requesting an *inter partes* review of claims 1–20 of U.S. Patent No. 11,811,582 B2 (Ex. 1001, the “’582 patent”).¹ Paper 3 (“Pet.”). Advanced Cluster Systems, Inc. (“Patent Owner”) timely filed a Preliminary Response. Paper 9 (“Prelim. Resp.”).

After the Institution Decision, Patent Owner filed a Response. Paper 15, (“Inst. Dec.”); Paper 21 (“Resp.”).² Then, Petitioner filed a Reply, and Patent Owner filed a Sur-reply. Paper 23 (“Reply”); Paper 24 (“Sur-reply”). Thereafter, the parties presented oral arguments in a consolidated oral hearing for IPR2025-00913–00916. *See* Paper 30 (“Tr.”).

We have jurisdiction under 35 U.S.C. § 6. This is a Final Written Decision under 35 U.S.C. § 318(a). Based on the full record, we determine that Petitioner demonstrates by a preponderance of the evidence that claims 1–20 of the ’582 patent are unpatentable.

A. Related Matters

The parties identify the following cases involving or related to the ’582 patent: *Advanced Cluster Sys., Inc. v. Intel Corp.*, No. 7:24-cv-00245 (W.D. Tex) and *Advanced Cluster Sys., Inc. v. Advanced Micro Devices*,

¹ The Board terminated Advanced Micro Devices, Inc. as a petitioner party to this *inter partes* review proceeding and in related proceedings IPR2025-00913–00915. Paper 20.

² The Acting Director denied Patent Owner’s request for discretionary denial and referred the Petition to the Board to “issu[e] a decision on institution addressing the merits and other non-discretionary considerations.” Paper 12, 3.

Inc., No. 7:24-cv-00244 (W.D. Tex.); *Advanced Cluster Systems, Inc. v. NVIDIA Corporation*, No. 1:19-cv-02032 (D. Del.) (dismissed); *NVIDIA Corp. v. Advanced Cluster Systems, Inc.*, IPR2021-00019 (PTAB) (institution denied); *NVIDIA Corp. v. Advanced Cluster Systems, Inc.*, IPR2021-00020 (PTAB) (institution denied); *Intel Corp. v. Advanced Cluster Systems, Inc.*, IPR2025-00794 (PTAB) (Director discretionary denial); *Intel Corp. v. Advanced Cluster Systems, Inc.*, IPR2025-00795 (PTAB) (Director discretionary denial); *Advanced Micro Devices, Inc. v. Advanced Cluster Systems, Inc.*, IPR2025-00862 (PTAB) (Director discretionary denial); *Advanced Micro Devices, Inc. v. Advanced Cluster Systems, Inc.*, IPR2025-00863 (PTAB) (Director discretionary denial); *Intel Corp. v. Advanced Cluster Systems, Inc.*, IPR2025-00913–915 (PTAB) (final written decisions determining all challenged claims unpatentable in related patents); *Ex parte Reexam. of U.S. Patent No. 10,333,768*, Control No. 90/015,706. Paper 5, 1–2; Paper 17, 1–2; Paper 22, 1–3; Pet. 116–17.

B. Real Parties-in-Interest

Petitioner identifies Intel Corp. and Advance Micro Devices, Inc. as real parties-in-interest. Pet. 116. Patent Owner identifies Advanced Cluster Systems, Inc. as the sole real party-in-interest. Paper 5, 1.

C. Summary of the '582 Patent³ (Ex. 1001)

The '582 patent, entitled “Cluster Computing,” describes “a computer cluster system comprises a plurality of nodes and a software package

³ The '582 patent issued on November 7, 2023, but has a series of continuation, continuation-in-part applications, and provisional applications

comprising a user interface and a kernel for interpreting program code instructions.” Ex. 1001, codes (54, 57). “[A] cluster node module is configured to communicate with the kernel and other cluster node modules” and “can accept instructions from the user interface and can interpret at least some of the instructions such that several cluster node modules in communication with one another and with a kernel can act as a computer cluster.” *Id.* at code (57).

Figure 1 of the '582 patent follows:

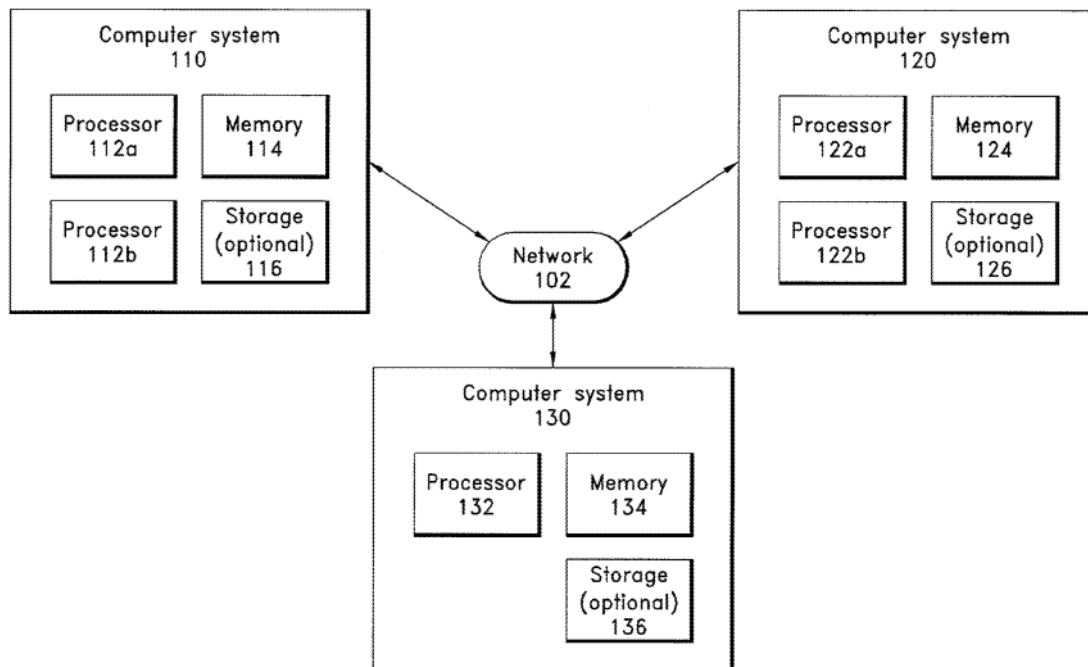


FIG. 1

Figure 1 “is a block diagram of one embodiment of a computer cluster 100 wherein computer systems 110, 120, 130 communicate with one another via a communication network 102.” Ex. 1001, 5:1–4.

60/813,738, filed on June 13, 2006, and 60/850,908 filed on Oct. 11, 2006. As such, the earliest priority date of the '582 patent is June 13, 2006.

In the embodiment shown in Figure 1, “a cluster system 100 includes computer systems 110, 120, 130 in communication with one another via a communications network 102.” Ex. 1001, 6:57–59. Each of the first, second, and third computer systems 110, 120, and 130 includes one or more processors, a memory device, and an optional storage device. *Id.* at 6:60–67. Further, each of the computer systems 110, 120, and 130 “includes a network interface (not shown) for connecting to a communications network 102, which can include one or more of a LAN, a WAN, an intranet, a wireless network, and/or the Internet.” *Id.* at 6:67–7:4.

Figure 2 of the '582 patent follows:

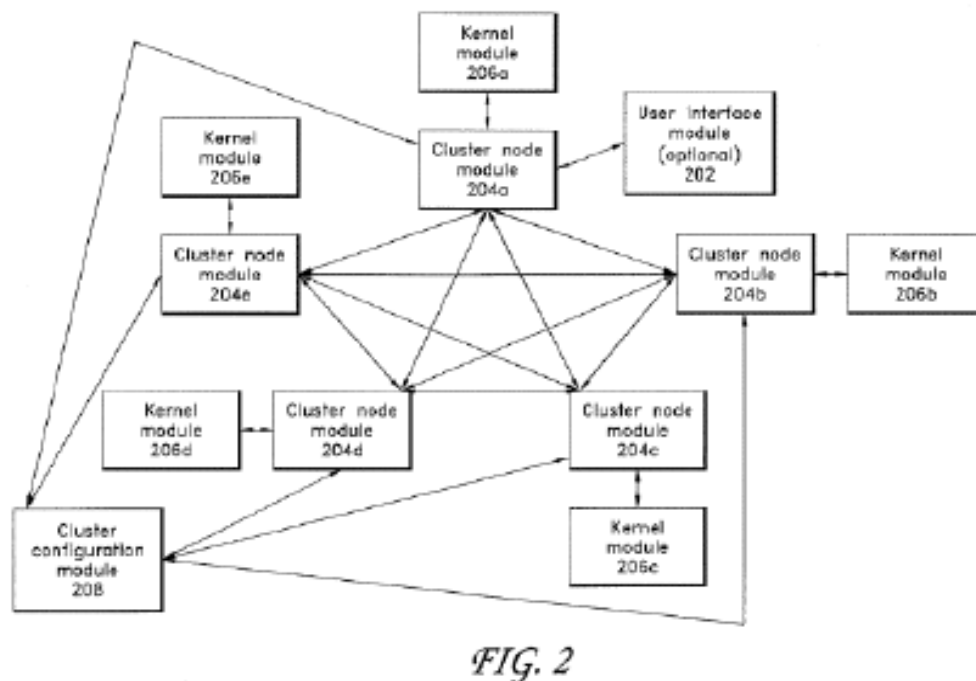


Figure 2 “is a block diagram showing relationships between software modules running on one embodiment of a computer cluster.” Ex. 1001, 4:6–8.

In the embodiments of Figures 1 and 2, “a cluster system 100 includes a user interface module 202 that is able to access a plurality of kernel [node]

modules 206a-e by communicating with a first cluster node module 204a.” Ex. 1001, 11:19–22. Kernel modules 206a–206e “can reside in the memory of one or more computer systems on which they run.” *Id.* at 11:27–29. Kernel modules 206a–206e “include single-threaded program code, [and] are each associated with one of the processors 112a, 112b, 122a, 122b, 132.” *Id.* at 11:34–36. Each of the kernel modules 206a–206e is in communication with a single cluster node module 204a–204e, respectively. *Id.* at 5:38–40. Cluster node module 204a is in communication with each of the other cluster node modules 204b–204e. *Id.* at 11:25–27. Communication between the cluster configuration module 208 and the cluster node modules 204a–204e initializes the cluster node modules 204a–204e to provide cluster computing support for the computer cluster 100. *Id.* at 11:41–44. Cluster node modules 204a–204e provide a way for many kernel modules 206a–206e such as, for example, Mathematica kernels, running on a computer cluster 100, to communicate with one another. *Id.* at 11:46–49. “Intercommunication provided by cluster computing permits exchange of information between nodes during the course of a parallel computation” to improve computational performance. *Id.* at 6:30–36.

D. Illustrative Claim

Independent claim 1, with Petitioner’s annotations, follows:

1. [1.0] A computer cluster comprising:
 - [1.1.1] a first processor;
 - [1.1.2] a second processor;
 - [1.1.3] a third processor;
 - [1.2] at least one computer-readable medium in communication at least one of the first processor, the second processor, or the third processor;

[1.3.1] a first kernel residing in the at least one computer-readable medium, [1.3.2] said first kernel configured to translate user commands into calls for execution of matrix operations on an array of data elements;

[1.4.1] a first cluster node module residing in the at least one computer-readable medium, [1.4.2] said first cluster node module configured to communicate with the first kernel and receive commands from a user interface;

[1.5.1] a second kernel residing in the at least one computer-readable medium, [1.5.2] said second kernel configured to execute of [sic] the matrix operations on the array of data elements and return a first result using the second processor;

[1.6.1] a second cluster node module residing in the at least one computer-readable medium, [1.6.2] said second cluster node module configured to send commands to the second kernel and [1.6.3] communicates with the first cluster node module;

[1.7.1] a third kernel residing in the at least one computer-readable medium, [1.7.2] said third kernel configured to receive the first result and execute additional matrix operations on the first and return a second result to one or more of the first, second, or third cluster node modules using the third processor; and

[1.8.1] a third cluster node module residing in the at least one computer-readable medium, [1.8.2] said third cluster node module configured to send commands to the third kernel and [1.8.3] configured to communicate with the first cluster node module and the second cluster node module

[1.9] wherein the first cluster node module is configured to return, based on the second result, a final result.

Ex. 1001, 29:18–56.

E. Asserted Grounds of Unpatentability

Petitioner asserts the following grounds of unpatentability (Pet. 14): ⁴

Claims Challenged	35 U.S.C. §	References/Basis
1–3, 5–19	103(a)	Menon, ⁵ Trefethen, ⁶ RS6000, ⁷ MPIref ⁸
4, 20	103(a)	Menon, Trefethen, RS6000, MPIref, POEref ⁹

Petitioner filed a Declaration of Chandrajit L. Bajaj, Ph.D. in support of the Petition. Ex. 1003. Patent Owner does not rely on expert testimony in support of the Response (or Sur-reply) “because it is not necessary for the Board to resolve the issues presented.” Resp. 9. Petitioner relies further on the Supplemental Declaration of Dr. Bajaj (Ex. 1163) and other declarations including the Declaration of expert librarian Ingrid Hsieh-Yee, Ph.D. (Ex.

⁴ The pre-AIA versions of 35 U.S.C. §§ 102, 103 apply because the challenged claims rely on an effective filing date (at the earliest) of June 13, 2006, which is prior to March 16, 2013, the effective date of the Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) (“AIA”). See Ex. 1001, code (63); Pet. 12.

⁵ “MultiMATLAB: Integrating MATLAB with High-Performance Parallel Computing,” Menon et al., SC ’97: Proceedings of the 1997 ACM/IEEE Conference on Supercomputing 1997. Ex. 1005.

⁶ “MultiMATLAB: MATLAB on Multiple Processors,” Trefethen et al., 1996 Computer Science Technical Report, Cornell University 1996. Ex. 1006.

⁷ “RS/6000 SP: Planning Vol. 1, Hardware and Physical Environment,” IBM 2001. Ex. 1007.

⁸ “MPI: A Message-Passing Interface Standard,” Message Passing Interface Forum 1994. Ex. 1017.

⁹ “Operation and Use, Volume, 1, Using the Parallel Operating Environment.” IBM 2001. Ex. 1008.

1086) in support of the Reply.

II. ANALYSIS

A. Overview

A claim is unpatentable under § 103 if the differences between the claimed subject matter and the prior art are such that the subject matter, as a whole, would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). The question of obviousness is resolved on the basis of underlying factual determinations, including (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art; (3) the level of skill in the art; and (4) when in evidence (not here), objective indicia of non-obviousness (i.e., secondary considerations). *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966).

A patent claim “is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art.” *KSR*, 550 U.S. at 418. An obviousness determination based on a combination of references requires finding “both [1] ‘that a skilled artisan would have been motivated to combine the teachings of the prior art references to achieve the claimed invention, and [2] that the skilled artisan would have had a reasonable expectation of success in doing so.’” *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1367–68 (Fed. Cir. 2016).

In an *inter partes* review, a petitioner bears the burden of persuasion to prove “unpatentability by a preponderance of the evidence.” *Dynamic Drinkware, LLC v. Nat'l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015) (quoting 35 U.S.C. § 316(e)). This burden of persuasion never shifts

to Patent Owner. *Dynamic Drinkware*, 800 F.3d at 1378 (citing *Tech. Licensing Corp. v. Videotek, Inc.*, 545 F.3d 1316, 1326–1327 (Fed. Cir. 2008)). Petitioner cannot satisfy its burden of proving obviousness by employing “mere conclusory statements,” but “must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.” *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016); *see also* 35 U.S.C. § 312(a)(3); 37 C.F.R. § 42.104(b)(4) (2024) (“The petition must specify where each element of the claim is found in the prior art patents or printed publications relied upon.”).

B. Level of Ordinary Skill in the Art

Determining whether an invention would have been obvious under 35 U.S.C. § 103 requires resolving the level of ordinary skill in the pertinent art at the time of the effective filing date of the claimed invention. *Graham*, 383 U.S. at 17. The person of ordinary skill in the art is a hypothetical person presumed to have known the relevant art at the time of the invention. *In re GPAC Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995). In determining the level of ordinary skill in the art, we may consider certain factors, including: “(1) the educational level of the inventor; (2) type of problems encountered in the art; (3) prior art solutions to those problems; (4) rapidity with which innovations are made; (5) sophistication of the technology; and (6) educational level of active workers in the field.” *Best Med. Int’l, Inc. v. Elekta Inc.*, 46 F.4th 1346, 1353 (Fed. Cir. 2022). The prior art of record may also reflect the level of ordinary skill in the art. *See Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001).

Petitioner, supported by Dr. Bajaj’s testimony (Ex. 1003 ¶¶ 24–27), submits that

[a] person of ordinary skill in the art (“POSITA”) in June 2006 would have been someone knowledgeable of and familiar with computer cluster systems and cluster computing techniques available at the time. Such a POSITA would have a bachelor’s degree in computer science, computer engineering, electrical engineering, or an equivalent training, and approximately two years of experience working in the field of cluster computing or parallel processing and would be knowledgeable regarding high-level scientific computing languages. Additional work experience can substitute for specific educational background, and vice versa.

Pet. 12–13 (citing Ex. 1003 ¶¶ 24–27).

Patent Owner does not oppose Petitioner’s proposed level of ordinary skill in the art. *See generally* Resp.

For the purposes of this trial, we adopt Petitioner’s undisputed level of ordinary skill in the art because it is consistent with the level reflected in the ’582 patent and the cited prior art.

C. Claim Construction

Petitioner notes that “[i]n IPR proceedings, claim terms receive their ‘ordinary and customary meaning[s],’” and “submit[s] that no claim term requires express construction” except for the limitation “peer-to-peer architecture” as recited in claim 15. *See* Pet. 12–13 (quoting 37 C.F.R. § 42.100(b)). Patent Owner generally agrees that “[c]laim terms are given their ordinary and customary meaning,” does not offer a claim construction, and contends that the “Response is direct[ed] to claims in which the Board did not construe any terms when granting institution” Resp. 9–10.

As Patent Owner generally contends, there is no reason to construe any terms to resolve the parties’ disputes in this proceeding. *See* Resp. 9–10, *Wellman, Inc. v. Eastman Chem. Co.*, 642 F.3d 1355, 1361 (Fed. Cir. 2011) (“[C]laim terms need only be construed ‘to the extent necessary to

resolve the controversy.”) (quoting *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999)).

D. Ground 1: Alleged Obviousness of Claims 1–3 and 5–19 over
Menon, Trefethen, RS6000, and MPIref

Petitioner contends that claims 1–3 and 5–19 would have been obvious over the combined teachings of Menon, Trefethen, RS6000, and MPIref. Pet. 23–110.

At the outset, Patent Owner argues that Petitioner fails to show that Menon, Trefethen, RS6000, MPIref, and POEref (asserted in Ground 2) “qualify as prior art” printed publications under 35 U.S.C. § 311(b). *See* Resp. 6, 40–46. In particular, Patent Owner asserts that the Petitioner has not established “public accessibility for” these references, and as a consequence, Petitioner’s “invalidity grounds fail.” *Id.*

A brief summary of each reference follows:

1. Menon (Exhibit 1005)

Menon is a technical article purportedly published by the IEEE Computer Society in connection with the ACM/IEEE SC ’97 Conference on Supercomputing 1997. Ex. 1005, 1. Menon’s title is “MultiMATLAB: Integrating MATLAB with High-Performance Parallel Computing.” *Id.*

Menon’s Abstract follows:

MATLAB is the most popular scientific computing environment available on uniprocessors today. Unfortunately, no such environment is currently available for multiprocessors. MultiMATLAB [1] is a general extension of the MATLAB environment to any distributed memory multiprocessors. This paper presents a new MultiMATLAB system designed to provide high-performance on multiprocessors while maintaining the functionality and usability of the MATLAB environment. This system will enable users to access high-performance parallel routines from within the MATLAB environment, to

extend the environment with new parallel routines, and to use these routines to develop parallel applications with the MATLAB language.

Id.

Figure 1 of Menon follows:

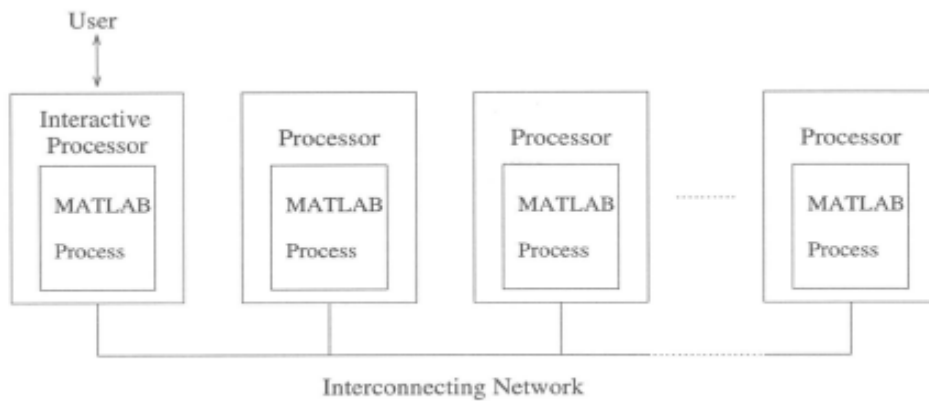


Figure 1: MultiMATLAB Architecture

Figure 1 is block diagram illustrating “MultiMATLAB Architecture.”

Ex. 1005, 3. Menon discloses that “each processor in a parallel platform individually runs a MATLAB process” and “[e]ach process is then provided with the ability to communicate with other processes through a communications layer that runs over the parallel platform’s interconnection network.” *Id.* Further, “[t]he user interacts directly with one MATLAB process, called the interactive process, and operates within that process’s MATLAB environment.” *Id.*

Figure 2 of Menon follows:

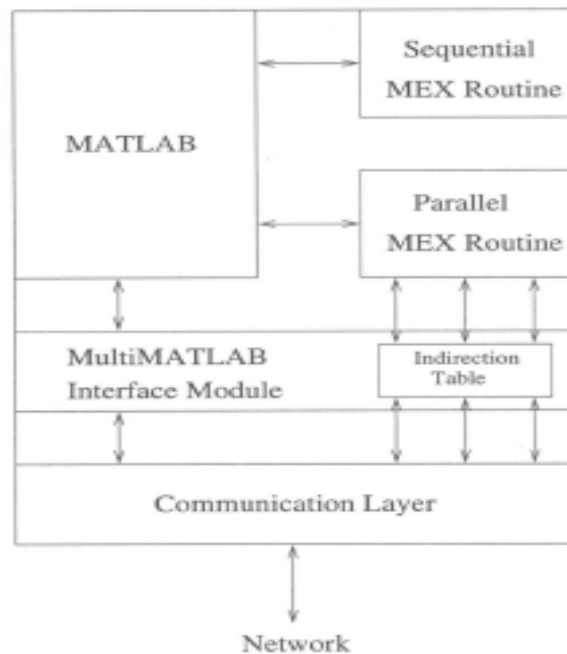


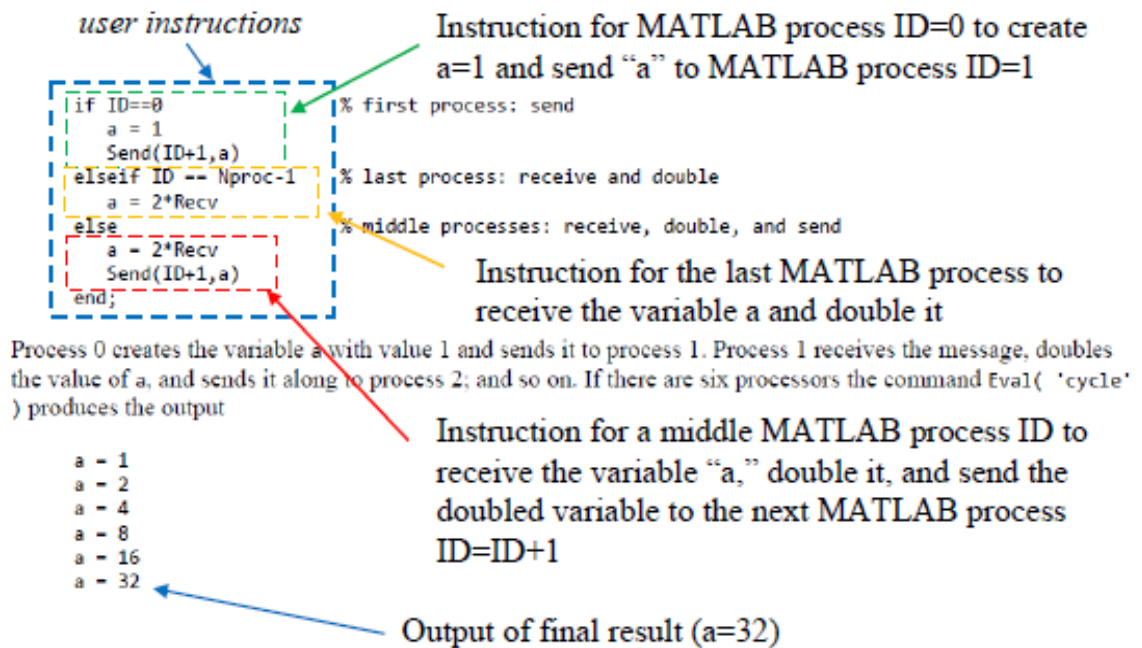
Figure 2: MATLAB Process Address Space

Figure 2 is a block diagram illustrating the “MATLAB Process Address Space.” Ex. 1005, 4. Figure 2 shows an interface module that “is responsible for initializing an underlying communication layer, such as MPI [2], PVM [3], BLACS [4], or any other package.” *Id.* at 3. Further “MEX routines, also shown in the Figure 2, are executables originally written in C or Fortran that may be run directly on MATLAB. All parallel functionality in the MULTIMATLAB system is provide through MEX routines.” *Id.*

2. Trefethen (Exhibit 1006)

Trefethen, titled “MultiMATLAB: MATLAB on Multiple Processors,” 1996 Computer Science Technical Report published by Cornell University 1996, pre-dates Menon (Ex. 1005) and discloses a system “that enables one to run MATLAB conveniently on multiple processors,” such as in the “IBM SP2,” “for fast and convenient execution of easily parallelizable

numerical computations on multiple processors.” Ex. 1006, 1 (Abstract); *see also id.* at 3–9. Trefethen discloses that “point-to-point communication is accomplished by send and receive commands.” *Id.* at 5–6. Trefethen teaches the use of “Send” and “Recv” commands to allow direct communication among nodes, as shown in annotated form in the Petition, reproduced below:



Pet. 21 (annotated Ex. 1006, 6). The sample code shown above illustrates the following node transactions:

- node 0 sending data “a = 1” to node 1,
- node 1 performing a mathematical calculation (multiply “a” by 2) and sending the result to node 2,
- node 2 performing a mathematical calculation (multiply “a” by 2) and sending the result to the next node (“and so on”), and
- the result (a=32) is output to the user.

Id. (citing Ex. 1003 ¶ 127; Ex. 1006, 6).

Trefethen further discloses a user “connected to a node of the IBM SP2 running MATLAB” and then “decid[ing] to start MATLAB on five new processors.” Ex. 1006, 3.

3. RS6000 (Exhibit 1007)

RS6000 is a chapter in an IBM publication for its RS6000 SP computer used in Menon’s system, entitled “RS/600 SP: Planning Vol. 1, Hardware and Physical Environment,” IBM 2001. Ex. 1007, 1–2.¹⁰ This document discloses “the processor nodes, model and expansion frames, the standard and optional hardware, and installation site planning . . . of an RS/6000 SP system.” *Id.* RS6000 states that “[t]he IBM RS/6000 SP System is scalable from one to 128 processor nodes” and that it uses “four, eight, twelve, or sixteen 375 MHz 630FP 64-bit processors per node.” *Id.* at 2, 9.

4. MPIref (Exhibit 1017)

MPIref, titled “MPI: A Message Passing Interface Standard,” Message Passing Interface Forum, 1994, and subtitled “Message Passing Interface Forum,” explains that “[t]he goal of the Message Passing Interface . . . is to develop a widely used standard for writing message-passing programs,” but it “is not sanctioned or supported by any official standards organization.” Ex. 1017, 1–2.¹¹

¹⁰ Citations are to the Exhibit page numbers added to the lower right corner of Exhibit 1017, rather than the internal page numbers of MPIref.

¹¹ Citations are to the Exhibit page numbers added to the lower right corner of Exhibit 1008, rather than the internal page numbers of POEref.

5. POEref (Ex. 1008)

POEref, titled “Operation and Use, Volume 1, Using the Parallel Operating Environment,” IBM 2001, is an IBM product manual for “the IBM Parallel Environment (PE)” and “its Parallel Operating Environment (POE)” running on an RS6000 system, as referenced in Menon. *See* Pet. 15 (citing Ex. 1003 ¶¶ 121–125; Ex. 1005, 6; Ex. 1008, 11; Ex. 1029; Ex. 1039).¹² POEref discloses an initialization process where a user enters the “poe” (parallel operating environment) command, such that “[w]hen you invoke poe, the Partition Manager allocates processor nodes for each task and initializes the local environment. It then loads your program, and reproduces your local environment, on each processor node.” Ex. 1008, 46.

6. Printed Publication Status of Petitioner’s Asserted Prior Art

The Federal Circuit instructs that “[t]o qualify as a printed publication, a reference ‘must have been sufficiently accessible to the public interested in the art.’” *Blue Calypso, LLC v. Groupon, Inc.*, 815 F.3d 1331, 1348 (Fed. Cir. 2016) (quoting *In re Cronyn*, 890 F.3d 1158, 1160 (Fed. Cir. 1989)). “A reference will be considered publicly accessible if it was ‘disseminated or otherwise made available to the extent that persons interested and ordinarily skilled in the subject matter or art exercising reasonable diligence, can locate it.’” *Id.* (quoting *Kyocera Wireless Corp. v. Int’l Trade Comm’n*, 545 F.3d 1340, 1350 (Fed. Cir. 2008)). Further, the question of whether a document qualifies as a “printed publication” for the purposes of 35 U.S.C. § 102 “involves a case-by-case inquiry into the facts and circumstances surrounding the reference’s disclosure to a member of the

¹² We cite to the exhibit page numbers added to the lower right corner of Exhibit 1008, rather than the internal page numbers of POEref.

public.” *Medtronic, Inc. v. Barry*, 891 F.3d 1368, 1380 (Fed. Cir. 2018) (citing *In re Klopfenstein*, 380 F.3d 1345, 1350 (Fed. Cir. 2004)).

The Petition assumes that “[t]he ’582 patent’s earliest possible priority date is June 13, 2006.” Pet. 12. Patent Owner does not dispute this date. *See generally* Resp. On this record, June 13, 2006, is the priority date for analysis.

For Exhibits 1005–1008 and 1017, Petitioner contends that each “is (i) more than 20 years old; (ii) is regular on its face with no signs of obvious alterations; and (iii) was found in a place where it could be expected to be found.” Pet. 16. Petitioner further contends that “each document is prima facie authenticated as an ancient document.” *Id.* (citing Fed. R. Evid. 901(8)). Patent Owner does not challenge the authenticity of any of these exhibits. *See generally* Resp. On this record, Petitioner establishes the authenticity of each exhibit.

Analysis of the parties’ arguments and evidence addressing the facts and circumstances surrounding the publication and public accessibility of the asserted references follows.

a. Overview of the evidence and arguments

The Petition provides substantial supporting evidence that Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), POEref (Ex. 1008), and MPIref (Ex. 1017) were publicly accessible to a person of ordinary skill in the art (POSITA) exercising reasonable diligence prior to June 13, 2006. Pet. 14–16. The Reply bolsters the Petition. Reply 15–22.

As indicated above, Patent Owner argues that Petitioner fails to establish public accessibility and, as such, “none of its references qualifies as prior art.” Resp. 40. For example, Patent Owner contends “Petitioner provides no evidence that a POSITA using typical search methods at the

time could have located [RS6000 and POEref (Exs. 1007–1008)].” *Id.* at 41. Patent Owner argues regarding these two IBM manuals that Petitioner fails to show a “link chain[.]” or web page that would allow “users to search for [the references] by topic” or “how the public could have navigated to the manuals.” *Id.* Patent Owner similarly argues that “Petitioner offers no evidence that anyone outside the MPI Forum ever knew of or accessed the draft [MPIref (Ex. 1017)].” *Id.* at 42–43. For the following reasons, Petitioner’s evidence is persuasive.

The publication evidence submitted with the Petition includes five declarations.¹³ Ex. 1003 (Bajaj Declaration) ¶¶ 121–123 (Menon), 127–128 (Trefethen), 130–131 (RS6000), 134–136 (POEref), and 140–142 (MPIref); Ex. 1027 (IEEE Declaration); Ex. 1029 (Internet Archive Declaration); Ex. 1047 (IBM Declaration); Ex. 1048 (Internet Archive Declaration); Ex. 1049 (exhibits to Ex. 1048).

For example, the IEEE Declaration of Gordon MacPherson (Ex. 1027) provides evidence concerning the publication and accessibility of the Menon reference (Ex. 1005). The Internet Archive Declaration of Mina Ching (Ex. 1029) provides evidence establishing the date of archived website URLs for accessing Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), POEref (Ex. 1008), and MPIref (Ex. 1017). The IBM Declaration of Carol Edwards (Ex. 1047) provides evidence regarding the publication and accessibility of POEref (Ex. 1008). The Internet Archive Declaration of

¹³ Patent Owner did not depose any of Petitioner’s declarants or submit rebuttal declarations in support of its Response. However, there is no requirement for Patent Owner to do so. That is, Petitioner bears the burden of showing unpatentability, including “prior art” printed publications.

Tanya Zeif (Ex. 1048) establishes the date of archived website URLs for accessing MPIref (Ex. 1017) on June 24, 1997 (Ex. 1049, 11)) and “Mathworks,” which provides a detailed discussion of MATLAB (March 31, 2004 (Ex. 1049, 248)). *See also* Ex. 1003 ¶¶ 141–142 (“Menon, while citing to MPIref, provided a website address of Argonne National Laboratory (<http://www.mcs.anl.gov/mpi>) for retrieving MPIref.” (citing Ex. 1005, 16)).

Patent Owner’s arguments do not undermine Petitioner’s evidence, which is specific, consistent, and persuasive in establishing public accessibility of the references prior to June 13, 2006. Petitioner is not required to prove a searchable “link chain,” contrary to Patent Owner’s arguments. *See* Resp. 41; Sur-reply 13–14, 17; *Medtronic*, 891 F.3d at 1380 (“Because there are many ways in which a reference may be disseminated to the interested public, ‘public accessibility’ has been called the touchstone in determining whether a reference constitutes a ‘printed publication’ bar under 35 U.S.C. § 102(b).” (citing *Blue Calypso*, 815 F.3d at 1348)). Petitioner’s burden is to establish by a preponderance of the evidence that each reference was publicly accessible to a POSITA exercising reasonable diligence prior to June 13, 2006. *Id.* Petitioner satisfies that burden.

Petitioner supports the Reply with the comprehensive expert librarian Declaration of Dr. Ingrid Hsieh-Yee. Ex. 1086; Ex. 1087 (Curriculum Vitae). Patent Owner objects to Petitioner’s reliance on Dr. Hsieh-Yee’s Declaration (Ex. 1085) in Reply as advancing a “new theories” of public accessibility. Sur-reply 14. Patent Owner further contends that “Reply evidence . . . may not establish a public-access channel the Petition never identified.” *Id.* (citing *VidStream LLC v. Twitter, Inc.*, 981 F.3d 1060, 1064–67 (Fed. Cir. 2020)). For the following reasons, Patent Owner’s “new theories” argument is unavailing.

In *VidStream*, the Federal Circuit rejected a similar argument that “the Board violated its own rules in considering evidence that was not provided with the IPR petitions, but only with the replies,” and that by relying on reply evidence the Board “departed from the specific grounds of unpatentability set forth in the petitions and thus exceeded its statutory authority.” *VidStream*, 981 F.3d at 1064, 1066. The *VidStream* court found nothing wrong in that case with the Board’s reliance on a declaration (by the same declarant involved here, Dr. Hsieh-Yee) submitted with petitioner’s reply brief, which cited Machine-Readable Cataloging (“MARC”) records of the type Petitioner relies on here to establish the date a reference was cataloged in a public library. *Id.* at 1063, 1066–1067; *see, e.g.*, Reply 17 (“MARC records” (Ex. 1091) “showing Menon was cataloged in the Academia Sinica Library on ‘December 14, 1998’” (citing Ex. 1086 ¶¶ 37–43)), 18 (“MARC records” (Ex. 1101) showing that Trefethen was cataloged by “Washington State University Library” on “November 3, 1997” (citing Ex. 1086 ¶¶ 79–85)). The *VidStream* court found “the Board acted appropriately, for the Board permitted both sides to provide evidence concerning the reference date of the [prior art] in pursuit of the correct answer.” *VidStream*, 981 F.3d at 1065.

Therefore, *VidStream* fully supports the Board’s reliance here on additional evidence of public accessibility submitted in support of Petitioner’s Reply, including Dr. Hsieh-Yee’s expert librarian Declaration (Ex. 1086).

Petitioner’s Reply evidence and arguments, moreover, do not constitute new theories, and the Board may consider such reply arguments and evidence when deciding the question of public accessibility. *See Rembrandt Diagnostics, LP v. Alere, Inc.*, 76 F.4th 1376, 1384–85 (Fed. Cir.

2023) (reply argument not improper if it “simply expands on previously raised arguments” or “is offered ‘to explain, repel, counteract, or disprove the evidence of the adverse party’”) (citations omitted). Petitioner’s Reply and Dr. Hsieh-Yee’s Declaration (Ex. 1086) are direct rebuttal to Patent Owner’s arguments. Consequently, Patent Owner’s “new theories” argument is unavailing.

For example, Petitioner replies to Patent Owner’s argument that “Petitioner has not proven that the SC’97 conference proceedings (including Menon) were actually disseminated to the public in 1997” (Resp. 45), by arguing “Menon qualifies as a printed publication because it was published and distributed at the Proceedings of the 1997 ACM/IEEE Conference on Supercomputing 1997.” Reply 15 (citing Ex. 1005; Ex. 1026; Ex. 1027). Petitioner further provides evidence of Menon being catalogued, indexed, and actually referenced in publications dated 1998, 1999, and 2000. *Id.* at 16–17 (citing, *e.g.*, Ex. 1086 ¶¶ 28–70). Petitioner’s Reply expands on the argument in the Petition by citing to Dr. Hsieh-Yee’s Declaration (Ex. 1086) to establish that “Menon was sufficiently disseminated and was located by POSITAs” who cited Menon in articles published prior to June 13, 2006. *Id.* at 15–16 (citing Ex. 1086 ¶¶ 54–63; Ex. 1094, 16, n.14; Ex. 1095, 19, n.2¹⁴; Ex. 1096, 14, n.17; Ex. 1097, 72).

Patent Owner’s Sur-reply does not challenge the facts established in Dr. Hsieh-Yee’s Declaration (Ex. 1086) with any cross-examination testimony of Dr. Hsieh-Yee. Sur-reply 15–21. Nor does Patent Owner cite inconsistent documentary evidence of record. *Id.*; *see also id.* at 22

¹⁴ Citations are to the Exhibit page number rather than the internal page number of the document.

(acknowledging that “Internet Archive captures of Menon (Ex. 1092) corroborate the same kind of evidence the Petition already relied on (Ex. 1029)”). Patent Owner’s position is that Petitioner’s evidence submitted in reply amounts to improper “new theories,” a position that is unfounded as set forth above and further below. *See* Sur-reply 13–16. Dr. Hsieh-Yee’s expert librarian declaration (Ex. 1086) is reliable and persuasive rebuttal testimony corroborated by documentary evidence of publication dates for each asserted reference prior to the June 13, 2006, priority date of the ’582 patent.

In summary, the record shows that the Reply properly responds to Patent Owner’s arguments by introducing rebuttal evidence that each prior art reference at issue was publicly accessible to a POSITA exercising reasonable diligence before June 13, 2006, and therefore qualifies as a prior art printed publication under 35 U.S.C. § 102.

Patent Owner also challenges the admissibility of the Petition’s publication evidence regarding all asserted prior art references and expresses an intent to “address those issues in a forthcoming motion to exclude.” Resp. 47. Motions to exclude were due July 23, 2026, but none were filed. Paper 16, 13. Accordingly, Patent Owner waives its right to challenge the admissibility of Petitioner’s evidence. 37 C.F.R. § 42.64(c) (“A motion to exclude evidence must be filed to preserve any objection.”).

We address specific publication evidence for each reference and additional Patent Owner arguments below.

b. Menon (Ex. 1005)

Petitioner contends that Menon “is a paper authored by Vijay Menon and Anne Trefethen of the Cornell research team” and “qualifies as a printed publication because it was published and distributed at the Proceedings of

the 1997 ACM/IEEE Conference on Supercomputing 1997.” Pet. 14 (citing Ex. 1005; Ex. 1026; Ex. 1027). Petitioner further contends that Menon was “published and distributed on Cornell’s website at least as early as 2001.” *Id.* (citing Ex. 1023; Ex. 1029). Petitioner next contends that “Menon was reasonably located by POSITAs because it had been cited 17 times in papers prior to the priority date of the ’582 patent.” *Id.* at 14 (citing Ex. 1003 ¶¶ 121–123; Ex. 1028, 4–9; Ex. 1029, 1–3).

The Declaration of Gordon MacPherson, Director, Board Governance & Policy Development at The Institute of Electrical and Electronics Engineers, Inc. (“IEEE”), establishes that Menon was publicly accessible to attendees of the IEEE SC ’97 conference in November 1997. Mr. MacPherson testifies as follows:

¶ 7. It is the regular practice of IEEE to publish articles and other writings including article abstracts and make them available to the public through IEEE Xplore. IEEE maintains copies of publications in the ordinary course of its regularly conducted activities.

¶ 8. The [Menon] article below has been attached as Exhibit A to this declaration:

¶ 9. I obtained a copy of Exhibit A through IEEE Xplore, where it is maintained in the ordinary course of IEEE’s business. Exhibit A is a true and correct copy of the Exhibit as it existed on or about March 4, 2025.

¶ 10. The article and abstract from IEEE Xplore show the date of publication. IEEE Xplore populates this information using the metadata associated with the publication.

¶ 11. [Menon] was published in SC ’97: Proceedings of the 1997 ACM/IEEE Conference on Supercomputing, held on November 15–21, 1997, in San Jose, CA, USA. Copies of the conference proceedings were made available no later than the last day of [the] conference. . . .

Ex. 1027, 1–2.

Patent Owner’s arguments that Petitioner does not show that Menon was “actually disseminated to the public in 1997” and Exhibit A to the MacPherson Declaration is “uncorroborated” by IEEE Xplore and “does not display any publication date for Menon,” are unavailing. Resp. 45–46. Mr. MacPherson’s unrebutted testimony is clear and specific: “[c]opies of the [SC ’97] conference proceedings were made available no later than the last day of conference,” i.e., no later than November 21, 1997. Ex. 1027 ¶ 11. Mr. MacPherson’s testimony is at least partially corroborated by the formal copyright notice and International Standard Book Number (“ISBN”) assigned to Menon and printed in the footer at the bottom of all 18 pages of Menon:

Proceedings of the ACM/IEEE SC97 Conference (SC’97)
0-89791-985-8/97 \$ 17.00 © 1997 IEEE

Ex. 1027, 4–21; *see also* Ex. 1005. The 1997 copyright date for the SC ’97 proceedings is consistent with the last three digits of the ISBN—“8/97”—indicating a date of August 1997.

Moreover, there is little doubt that Menon was publicly accessible before the June 13, 2006, priority date of the ’582 patent, the legally relevant consideration, given Dr. Hsieh-Yee’s authoritative and unrebutted testimony:

The ACM metadata record shows *Menon* was published on November 15, 1997. . . . [L]ibrary records inform my opinion that the conference proceedings containing *Menon* became publicly discoverable and accessible in libraries no later than December 1998. . . .

Public access to the conference proceedings containing *Menon* was actually earlier than July 1998. The General Information webpage of the SC’97 conference website, archived

on February 26, 1998 (Ex. 1093), informed users that the conference proceedings [were] available online, and *interested users could use workstations at the conference to view the conference proceedings during the November 1997 conference. These dates inform my opinion that the SC'97 conference proceedings containing Menon was publicly accessible as early as November 1997, and no later than February 1998* when the webpage was archived by Internet Archive. Copies of the conference main page, table of contents, technical papers, and abstracts of technical papers archived in April 1998 (included in Ex. 1093), confirm that *Menon* was included in the SC'97 conference, and the conference proceedings [were] publicly discoverable on the Internet no later than April 1998.

Ex. 1086 ¶¶ 65, 67 (emphasis altered); *see also id.* ¶¶ 28–70 (establishing authenticity and public availability of Menon).

The Internet Archive Affidavit of Mina Ching further supports Petitioner's contention that Menon was publicly available on the Cornell website prior to June 13, 2006. Ex. 1029. The Ching Affidavit states that the copy of Menon attached as Exhibit A is a “true and accurate cop[y] of browser screenshots . . . of the archived files for the URLs and the dates specified in the attached coversheet of each screenshot.” *Id.* ¶ 6. The dates shown on Exhibit A to the Affidavit indicate that Menon was available on the Cornell website no later than November 18, 2003. *Id.* (Ex. A at 5–10). Patent Owner's argument, that “Petitioner offers no evidence that the Menon PDF was indexed or that anyone actually located it on Cornell's site within the relevant timeframe” does not undermine the Petition's contention of public accessibility, at least to the extent each archived link was accessible

before June 13, 2006.¹⁵ *See* Resp. 46. Moreover, Petitioner contends that Menon “has been cited 17 times in papers with online publication dates between the original publication of Menon and the earliest priority date of the ’582 patent.” Ex. 1003 ¶ 123 (citing Ex. 1028, 4–9). At least 14 of the linked papers of Exhibit 1028 have online publication dates before the earliest possible effective filing date of the ’582 patent. *See* Exhibit 1028, 5–10. The Internet Archive Affidavit of Mina Ching (Ex. 1029) and the Declaration of Dr. Bajaj (Ex. 1003) provide persuasive support for Petitioner’s argument.

For the reasons given above, we determine Petitioner establishes by a preponderance of the evidence that Menon was publicly accessible before June 13, 2006, and qualifies as a prior art printed publication under 35 U.S.C. § 102.

c. Trefethen (Ex. 1006)

Petitioner contends that Trefethen “qualifies as a printed publication because it was published and distributed on the webpage of one of its authors at least as early as 2001” and “was also published on a Cornell University website no later than May 10, 1996.” Pet. 15 (citing Ex. 1023; Ex. 1029). Petitioner also contends that Trefethen was available publicly because “it has been cited 10 times in papers prior to the priority date of the ’582 patent.” *Id.* (citing Ex. 1003 ¶¶ 129–130; Ex. 1035, 2–3). The Petition

¹⁵ For example, the link on page 5 of Exhibit 1028 leads to an article dated March 15, 2006, that cites Menon in footnote 19. Ex. 1028, 5 (Karpievitch Y and Almeida J. (2006). “*mGrid: A load-balanced distributed computing environment for the remote execution of the user-defined Matlab code.*” BMC Bioinformatics. 10.1186/1471-2105-7-139. 7:1. Online publication date: 15-Mar-2006, available at <https://doi.org/10.1186/1471-2105-7-139>). The earliest citation to Menon is July 1, 1998. *Id.* at 7 (n.5 in linked article).

further contends that a “POSITA seeking to understand MultiMATLAB’s capabilities would have consulted the Cornell website dedicated to MultiMATLAB, which contained links to [Trefethen and Menon].” *Id.* at 24 (citing Ex. 1003 ¶ 82; Ex. 1023).

Patent Owner responds that “Petitioner offers no competent evidence that . . . a hyperlink existed or that anyone actually found the page in the 1990s.” Resp. 43 (citing Ex. 1023). Patent Owner also argues that “[t]he record contains no proof of indexing, public discoverability, or that the site was accessed by any member of the public,” such that “one would have needed to know the exact URL to load the page – which is not public accessibility unless that URL was advertised or indexed somewhere.” *Id.* Patent Owner further emphasizes that “it is undisputed that before the June 2006 critical date, Trefethen had not been formally published in any journal or conference, nor is there evidence it was available in libraries. Petitioner relies entirely on alleged web availability.” *Id.* at 44 (citing *Blue Calypso*, 815 F.3d at 1349).

First, although indexing and cataloging are important considerations, the lack of indexing and cataloging is not dispositive. As the Federal Circuit explained in *Blue Calypso*, “we found that a particular article that was available only through an on-line publication was publicly accessible . . . [because] the particular on-line publication was well known to the community interested in the subject matter of the reference.” *Blue Calypso*, 815 F.3d at 1349 (citing *Voter Verified, Inc. v. Premier Election Sols., Inc.*, 698 F.3d 1374, 1380 (Fed. Cir. 2012)). Here, Dr. Bajaj testifies as follows:

During the relevant timeframe, researchers commonly made their findings available on websites, particularly the websites associated with their research institution, here Cornell University. *A POSITA working with the MultiMATLAB system*

would have consulted the Cornell website dedicated to MultiMATLAB, which contained links to both papers. Co-author Vijay Menon likewise included links to both papers within his web page on the Cornell website. Furthermore, the authors “Vijay Menon” and “Anne E. Trefethen” are common across both papers. The authors Menon and Trefethen were both at the same university, Cornell University, *working on the same research team to develop the MultiMATLAB system.* Accordingly, any POSITA seeking to understand MultiMATLAB would have been presented with both papers to consider.

Ex. 1003 ¶ 153 (citing Ex. 1005, 1; Ex. 1006, 1; Ex. 1023, 1–41) (emphases added). Dr. Bajaj continues:

Cornell was recognized at the relevant timeframe as a *pioneer in ‘the development of high-performance “cluster” computers* made by linking off-the-shelf computers in parallel,” and an ordinarily skilled artisan “interested in computer cluster systems and cluster computing techniques and exercising reasonable diligence would have located Trefethen either from the Cornell website and/or the ACM Digital Library, as evidenced by the afore-mentioned 10 citations to Trefethen.”

Ex. 1003 ¶ 130 (quoting Ex. 1033, 2; citing Ex. 1034, 1) (emphasis added). Dr. Bajaj’s testimony is consistent with the evidence of record, unrebutted, and aligns with the Federal Circuit’s reasoning in *Blue Calypso*. *See Blue Calypso*, 815 F.3d at 1350 (“[A] published article with an express citation to the potentially invalidating reference would similarly provide the necessary guidance”).

Patent Owner does not offer cross-examination or rebuttal testimony of Dr. Bajaj. Dr. Bajaj’s testimony that Cornell was a recognized leader in the field of high-performance cluster computing during the timeframe in question is bolstered by the multiple prior art citations to Menon and Trefethen prior to the ’582 patent’s priority date. We credit Dr. Bajaj’s testimony that, given the prominence of the Cornell research team in the

MATLAB multi-processor art, a POSITA exercising reasonable diligence easily could have located Trefethen from the Cornell website or ACM Digital Library in order to explore MATLAB multi-processor art.

Dr. Hsieh-Yee adds further weight to Petitioner’s argument. Dr. Hsieh-Yee testifies that Trefethen was disseminated and actually located by POSITAs:

Based on the records and documents I have reviewed, it is my opinion that *Trefethen* was published in May 1996, and was archived by Internet Archive on May 10, 1996, making this work discoverable on the Internet on that date. In 1996, 1997 and beyond, interested users could use search engines to search for MultiMATLAB or MATLAB to discover *Trefethen*. They could also use search engines to discover the lead author’s webpage and follow the link for MultiMATLAB on that webpage to access *Trefethen*. The Washington State University Library cataloged *Trefethen* on November 3, 1997, making this work discoverable at this library on that date. *Three early citations were published in 1997 and 1998, and became publicly accessible at libraries in 1998 and 1999.* The earliest citation is a dissertation completed and approved in September 1997.

Ex. 1086 ¶ 104 (emphasis added); *see also id.* ¶¶ 76–104. As Petitioner argues, a preponderance of the evidence establishes that “Trefethen was cataloged, indexed, and read and used by POSITAs” prior to June 13, 2006. Reply 18 (citing Ex. 1023; Ex. 1029; Ex. 1035, 2–3; Ex. 1003 ¶¶ 127–128; Ex. 1086 ¶¶ 71–104).

Patent Owner notes there is “an instance of self-citation” in Menon but argues “[c]itation alone does not prove that the public could locate [Trefethen].” Resp. 44 (citing Ex. 1005, 1–2). However, Patent Owner does not dispute that “Trefethen was ‘cited 10 times’ before the relevant date.” *Id.* Patent Owner’s argument is weak and unavailing when weighed against the substantial evidence presented by Petitioner. Moreover, even without a

known URL or hyperlink to Trefethen, Menon's and other prior art's citations to Trefethen provide additional motivation for an interested artisan to pursue finding Trefethen on the Cornell website or ACM Digital Library by exercising reasonable diligence to supplement the teachings of Menon and/or parallel MultiMATLAB teachings in general. Accordingly, Petitioner establishes by a preponderance of the evidence that a POSITA exercising reasonable diligence would have located Trefethen from the Cornell website or ACM Digital Library prior to June 13, 2006.

For the reasons given above, we determine Petitioner establishes by a preponderance of the evidence that Trefethen was publicly accessible before June 13, 2006, and qualifies as a prior art printed publication under 35 U.S.C. § 102.

d. RS6000 (Ex. 1007), POEref (Ex. 1008), and MPIref (Ex. 1017)

First, Petitioner contends that RS6000 (Ex. 1007) "is a product manual by IBM in support of the IBM RS/6000 SP computing solution." Pet. 15. According to Petitioner, "RS6000 qualifies as a printed publication because it was cataloged and indexed in IBM's online document library at least as early as 2002 on IBM's website." *Id.* (citing Ex. 1003 ¶¶ 132–133; Ex. 1007; Ex. 1029; Ex. 1038). Petitioner also contends that "Menon describes the MultiMATLAB architecture on an 'IBM SP2' and RS6000 provides details of that system." *Id.* at 26 (citing Ex. 1005, 5–6); *id.* at n.2 (explaining that "'IBM SP2' is the same as 'IBM RS/6000 SP.'" (citing Ex. 1010, 6 ("[T]he SP2 was renamed to simply the SP."))). Petitioner further contends that a "POSITA seeking to understand MultiMATLAB would be motivated to seek the documentation corresponding to the systems for which 'MultiMATLAB was design[ed] specifically.'" *Id.* (quoting Ex. 1005, 6).

Second, Petitioner contends that POEref (Ex. 1008) “is a product manual published by IBM in support for the Parallel Operating Environment running on an RS/6000 system.” Pet. 15. According to Petitioner, “POEref qualifies as a printed publication because it was catalogued and indexed in IBM’s online document library at least as early as 2001” and “was distributed with the IBM SP2 systems and published at least as early as 2001.” *Id.* at 15–16 (citing Ex. 1003 ¶¶ 134–136; Ex. 1008, 157; Ex. 1036, 14; Ex. 1047). Dr. Bajaj testifies that “the IBM SP2 parallel computing platform was distributed with manuals, including the ‘IBM AIX Parallel Environment—Operations and Use’ manuals.” Ex. 1003 ¶ 137 (citing Ex. 1036, 14; Ex. 1008, 157–159 (stating “PE documentation is shipped with the PE licensed program in a variety of formats” and indicating “IBM Parallel Environment for AIX: Operation and Use, Volume 1, SA22-7425” (Ex. 1008) as one of the PE publications). Menon describes the IBM SP2 as “a modern high performance distributed memory multiprocessor,” with “IBM’s proprietary version of MPI specifically optimized for the SP2.” Ex. 1005, 5.

Third, Petitioner contends that “MPIref qualifies as a printed publication because it was distributed to POSITAs using FTP mail servers by the University of Tennessee and Oak Ridge National Library in 1994.” Pet. 16 (citing Ex. 1017, cover page; Ex. 1029, 1–3). Petitioner also contends that MRIref (Ex. 1017) “was published on MPI Forum’s website no later than July 3, 1998” and “was reasonably located by POSITAs because it had been cited 74 times in papers with online publications dates before the earliest priority date of the ’582 patent.” *Id.* (citing Ex. 1003 ¶¶ 140–142; Ex. 1037, 2–16). Petitioner further contends that Menon “expressly cite[s] to ‘the MPI communication standard’” and “MPIref is the

MPI communication standard referenced by Menon.” *Id.* at 27 (quoting Ex. 1005, 5; citing Ex. 1017, 1 (“MPI: A Message-Passing Interface Standard”)).

Patent Owner’s arguments contesting the public accessibility of RS6000 (Ex. 1007), POEref (Ex. 1008), and MPIref (Ex. 1017) essentially repeat similar arguments with respect to Menon (Ex. 1005) and Trefethen (Ex. 1006), are largely addressed above, and are unavailing. *See* Resp. 41–42 (asserting Petitioner’s link chains are not sufficient).

As summarized above, the Petition presents persuasive evidence that artisans of ordinary skill were aware of IBM SP2 and POE products at least through publication of Menon (Ex. 1005) and via distribution of POEref (Ex. 1008) with the IBM SP2 product. *See* Pet. 14–15 (citing Ex. 1003 ¶¶ 132–133, 134–136, 140–142; Ex. 1007; Ex. 1008, 157; Ex. 1029; Ex. 1036, 14; Ex. 1038; Ex. 1039; Ex. 1047), 25–26 (citing Ex. 1005, 5). Patent Owner agrees that “Petitioner’s evidence shows the manuals were on IBM’s server by 2005,” but asserts “they were [not] searchable via internet engines.” Resp. 42. Patent Owner also acknowledges that “Petitioner notes the manuals shipped with RS/6000/SP2 systems.” *Id.*

Patent Owner’s arguments support Petitioner’s showing that artisans of ordinary skill reading Menon (Ex. 1005) would have been aware of IBM systems involving MATLAB in general and would have been motivated to search for products and systems described in Menon (Ex. 1005) and obtain the manuals at issue here for further implementation instruction regarding MATLAB.

POEref (Ex. 1008) provides specific instructions for locating “IBM product information for RS/6000 SP products on the World Wide Web,” thereby indicating an intent to distribute the manuals:

Most of the RS/6000 SP hardware and software books are available from the IBM Web site at:

<http://www.ibm.com/servers/eserver/pseries>

The serial and parallel programs that you find in the *IBM Parallel Environment for AIX: Hitchhiker's Guide* are also available from this Web site, in the same location as the PE online library.

You can view a book, download a Portable Document Format (PDF) version of it, or download the sample programs from the *IBM Parallel Environment for AIX: Hitchhiker's Guide*.

At the time this manual was published, the Web address of the *RS/6000 SP Product Documentation Library* page was:

http://www.rs6000.ibm.com/resource/aix_resource/sp_books

Ex. 1008, 157.

Another statement in POEref (Ex. 1005) further indicates an intent by IBM to disseminate that manual to a wide variety of artisans of ordinary skill:

This book is designed primarily for end users and application developers. It is also intended for those who run parallel programs, and some of the information covered should interest system administrators. Readers should have knowledge of the AIX operating system and the X-Window system. Where necessary, this book provides some background information relating to these areas. More commonly, this book refers you to the appropriate documentation.

Ex. 1008 ix.

Dr. Hsieh-Yee testifies as follows:

My research found that archived webpages on the IBM website show *POE-2001* was available no later than February 2002 and continued to be available beyond early 2004. Internet Archive does not archive every web file or resource, and it has not archived *POE-2001*. Changes in IBM servers and system architecture may have contributed to this situation. I have obtained several archived webpages from the IBM website to show how interested users could discover *POE-2001* in 2002,

2003, and 2004, and how this document was presented as available for online browse or download on IBM webpages in those years. These webpages are presented in Ex. 1113 in this declaration. In addition, two citing documents are included to show *POE-2001* was known and used by researchers in 2004 and 2005.

Ex. 1086 ¶ 128. Dr. Hsieh-Yee’s testimony is unrebutted.

Further as to POEref, the Petition shows Menon teaches “that ‘POE . . . is used to start and assign identification numbers to all the MATLAB processes when implementing the MultiMATLAB architecture in IBMSP2, and POEref is directed to describ[ing] . . . [the] Parallel Operating Environment (POE),” and an ordinarily skilled artisan “seeking to understand MultiMATLAB would be motivated to review the documentation corresponding to the systems expressly discussed in Menon.” Pet. 111 (citing Ex. 1005, 5–6; Ex. 1008, ix).

Based on the foregoing discussion, the Petition adequately supports the contention that RS6000 (Ex. 1007) and POEref (Ex. 1008) were “sufficiently accessible to the public interested in the art” including via the links Petitioner cites, in order to implement known MultiMATLAB products including the known POE system for which the known IBM SP2 was designed. *See Blue Calypso*, 815 F.3d at 1348; Ex. 1005, 5; Ex. 1008, 157. Dr. Hsieh-Yee’s Declaration credible and unrebutted testimony further supports Petitioner’s showing. *See* Pet. 107–108; Reply 21–22 (citing Ex. 1086 ¶¶ 105–148).

Turning to MPIref (Ex. 1017), Petitioner further shows that “Menon indicates that a POSITA would look to use the MPI standard because of MPI’s ‘broad popularity, its suitability to high performance parallel computing, and the virtually universal availability of vendor-optimized

versions on modern parallel computers.” Pet. 27 (quoting Ex. 1005, 5 (“IBM’s proprietary version of MPI [is] specifically optimized for the SP2”)). This evidence further supports the Petition’s contention that an artisan of ordinary skill would have been motivated and able to locate MPIref (Ex. 1017) as documentation of IBM’s optimized MPI for use in the SP2 system.

Patent Owner contends that “Petitioner offers no evidence that anyone outside the MPI Forum ever knew of or accessed the draft [MPIref].” Resp. 42–43. Petitioner replies with authoritative evidence from Dr. Hsieh-Yee:

Based on the documents and records I have reviewed, it is my opinion that *MPI-1994* became publicly accessible by November 1994 when a citing document was published, and when the National Science Library of Canada created their original MARC record for this standard. *MPI-1994* was definitely available no later than July 1998 when it was first archived by Internet Archive.

Ex. 1086 ¶ 182; Reply 23–24 (citing Ex. 1086 ¶¶ 149–182).

Patent Owner’s unsupported Sur-reply contention, that “Petitioner offers no contemporaneous evidence that any skilled person actually accessed” the references “through the cataloging records or citation chains the Reply now invokes” (Sur-reply 20–21), is unavailing because it does not undermine the substantial evidence of record satisfying Petitioner’s burden of proof by a preponderance of the evidence that MPIref was publicly accessible to a POSITA exercising reasonable diligence prior to June 13, 2006.

Moreover, Petitioner’s evidence further shows that MPIref (Ex. 1017), a recognized communication standard, “was reasonably located by POSITAs because it had been cited 74 times in papers” prior to the priority date of the ’581 patent. See Pet. 16 (citing Ex. 1037, 2–16; Ex. 1003 ¶¶ 142–144. For

the reasons similar to those discussed in connection with Menon (Ex. 1005) above, Petitioner's evidence persuasively shows public availability.

In summary, the evidence cited with the Petition and further supported by the Reply evidence shows an intent to distribute RS/6000 (Ex. 1007), MPIref (Ex. 1017), and POEref (Ex. 1008) via the IBM website, product shipments, or otherwise, to allow users to successfully implement the known products using well-known standards associated therewith. *See Jockmus v. Leviton*, 28 F.2d 812 (2nd Cir. 1928) (“[T]he catalogue itself . . . bear[s] its own evidence of existence since 1908, but no one can seriously suppose that such a document, printed in quantity, was intended to be kept secret; its whole purpose was to be spread broadcast as far as possible. It had been printed at some expense in French for French customers, and, unless some accident happened to prevent, it would in due course have gone upon its intended errand.”) (Judge L. Hand).

For the reasons given above, we determine Petitioner establishes by a preponderance of the evidence that RS6000 (Ex. 1007), MPIref (Ex. 1017), and POEref (Ex. 1008) were publicly accessible before June 13, 2006, and qualify as a prior art printed publication under 35 U.S.C. § 102.

e. Conclusion as to Public Accessibility

Based on the foregoing discussion, we determine Petitioner establishes by a preponderance of the evidence that Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), POEref (Ex. 1008), and MPIref (Ex. 1017) were publicly accessible prior to June 13, 2006, and qualify as prior art printed publications under 35 U.S.C. § 102.

7. Analysis of Claim 1

Petitioner contends that claim 1 would have been obvious over the combined teachings of Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), and MPIref (Ex. 1017). Pet. 23–67.

Petitioner presents arguments and evidence concerning why an ordinarily skilled artisan would have been motivated to combine Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), and MPIref (Ex. 1017) with a reasonable expectation of success. *Id.* at 23–28. Petitioner also maps the prior art to the limitations of claim 1. *Id.* at 27–67.

Patent Owner argues that Petitioner fails to establish a motivation to combine Menon and Trefethen, that Menon teaches away, and that Petitioner fails to establish some claim limitations. PO Resp. 12–16, and 22–27, 29–33.

a. Motivation to Combine Menon (Ex. 1005) and Trefethen (Ex. 1006)

Petitioner contends that “Menon suggests the combination by expressly citing Trefethen and explaining that Menon builds on the teachings of Trefethen.” Pet. 24 (citing Ex. 1005, 1–2).

Petitioner next contends that “Trefethen describes ‘how the [MultiMATLAB] system is used’ using code examples that perform parallel mathematical calculation using MATLAB” and “Menon describes ‘tak[ing] advantage of high performance distributed memory processors’ to enable MATLAB to perform mathematical calculations.” Pet. 24 (quoting Ex. 1005, Abstr.; Ex. 1006, 2; citing Ex. 1005, 16). According to Petitioner, “Menon and Trefethen together represent a combination of prior art elements according to known methods (executing MATLAB code on a multiprocessor system) to yield the predictable result discussed in both

papers: the MultiMATLAB system executing MATLAB in parallel to perform mathematical calculations.” *Id.* (citing Ex. 1005, Abstr.; Ex. 1006, 2–7).

Petitioner also contends that combining “Menon and Trefethen to parallelize MATLAB would have been predictable because researchers had a strong desire to parallelize MATLAB,” “[t]he goal of parallelizing MATLAB was ‘inevitable,’” and the combination “achieved this goal: the MultiMATLAB system executes MATLAB in parallel to perform mathematical calculations.” Pet. 25 (citing Ex. 1003 ¶¶ 151–154; Ex. 1005, 2; Ex. 1006, 11–12; quoting Ex. 1006, 12).

Patent Owner responds that Petitioner’s argument “reduces to the unremarkable proposition that researchers wanted to parallelize high-level mathematical computing environments,” which is an insufficient reason to combine the references and fails to “explain why a POSITA would have selected and implemented the particular multicore, result-return, and ordered sequence architecture recited in the claims.” Resp. 24.

Patent Owner also argues that the combination is based on impermissible hindsight. Resp. 27–29. Patent Owner further generally argues that Menon (Ex. 1005) refutes Petitioner’s premise that Menon (Ex. 1005) is “an incremental extension of Trefethen” because Menon “distinguishes the first MultiMATLAB architecture (Trefethen) from the newer one, describing the first as ‘different’ and identifying performance and extensibility limitations that prompted a new design.” *Id.* at 25 (citing Ex. 1005, 14), 26 (“Menon identifies structural limits that prevent the first architecture (Trefethen) from supporting the kinds of extensions and performance that users wanted.”).

In the Reply, Petitioner argues persuasively that Patent Owner ignores the multiple reasons set forth in the Petition that Dr. Bajaj’s expert testimony supports as to why an ordinarily skilled artisan would have been motivated to combine Menon and Trefethen. Reply 14–15 (citing Pet. 23–24; Ex. 1003 ¶¶ 75–83; Ex. 1005, Abstract, 1–2; Ex. 1006, 2–7; Ex. 1163 ¶¶ 33. Petitioner further argues persuasively that “Patent Owner’s position is entirely attorney argument with no supporting expert testimony.” *Id.* at 13 (citing Ex. 1163 ¶¶ 37–39).

Moreover, Petitioner bolsters the showing in the Petition in response to Patent Owner’s arguments by providing further testimony from Dr. Bajaj with the Reply (which essentially supports and builds on previous testimony provided with the Petition) that a POSITA would have been motivated to combine Menon (Ex. 1005) and Trefethen (Ex. 1006) in the manner claimed in the ’582 patent:

Menon suggesting the combination by **expressly citing** Trefethen and explaining that Menon builds on the teachings of Trefethen. Ex.1005, 1–2; Ex.1003, ¶71.

Menon and Trefethen both describe aspects of how the same MultiMATLAB system enables MATLAB to perform parallel mathematical calculations. Ex.1005, Abstract, 16; Ex.1006, 2; Ex.1003, ¶72.

Menon and Trefethen together represent a combination of prior art elements according to known methods (executing MATLAB code on a multi-node system) to yield the predictable result discussed in both references: the MultiMATLAB system executing MATLAB in parallel to perform mathematical calculations. Ex.1005, Abstract; Ex.1006, 2–7; Ex.1003, ¶70.

Reply 14 (citing Ex. 1163 ¶ 33).

Dr. Bajaj testifies credibly as follows:

A POSITA would have had an especially strong reason to read these references together *because they address the same*

project, the same MATLAB-based parallel-computing problem, and the same MultiMATLAB command environment. Ex. 1005, 1–2; Ex. 1006, 2–7. Menon expressly identifies Trefethen’s MultiMATLAB work and then describes a redesigned architecture that preserves the same user-level objective—parallel MATLAB computations—*while improving implementation details and extensibility.* Ex. 1005, 2, 14.

A POSITA therefore would not view the references as mutually exclusive alternatives. Rather, a POSITA would understand Trefethen as teaching specific MultiMATLAB commands and examples, including Eval, Send, Recv, and the cycle example, and Menon as teaching *how the MultiMATLAB architecture was implemented and improved in a later version.* Ex. 1005, 2–4, 14; Ex. 1006, 4–7.

Ex. 1163 ¶¶ 37–38 (emphases added).

On this record, Petitioner and Dr. Bajaj demonstrate a sound technical motivation for a POSITA to combine Menon (Ex. 1005) and Trefethen (Ex. 1006), i.e., to improve MATLAB’s operational performance and extensibility in the same parallel operating environment. That is, Menon expressly builds on Trefethen to improve and extend the parallel processing capabilities of MultiMATLAB: “This system will enable users to access high-performance parallel routines from within the MATLAB environment, *to extend the environment with new parallel routines, and to use these routines to develop parallel applications with the MATLAB language.*” Ex. 1005, Abstr. (emphasis added). By citing Trefethen, Menon invites artisans to read Menon in conjunction with Trefethen, as is the natural result expected when two references generated by the same research group at the same university address the same technical subject matter about a year apart. Ex. 1005, 1; Ex. 1006, Ex. 1086 ¶¶ 65, 101, 102. The first citation in Menon is to Trefethen, underscoring its importance in understanding the Menon system. Ex. 1005, 2, 16.

In contrast, Patent Owner’s Response and Sur-reply assert a lack of motivation to combine references based on mere unsupported attorney argument. *See* PO Resp. 22–26; Sur-reply 16–20. Patent Owner contends Petitioner’s argument “reduces to the unremarkable proposition that researchers wanted to parallelize high-level mathematical computing environments,” which Patent Owner asserts is an insufficient reason to combine the references because “[m]otivation must be tied to the claimed configuration, not to an abstract goal.” Resp. 24 (citing *KSR*, 550 U.S. at 418).

This line of argument is unavailing because there is nothing abstract about the stated goal of parallelizing high-level mathematical computing environments using MATLAB in particular. Patent Owner’s arguments overlook Petitioner’s evidence that Menon specifically references, builds on, and expands the teachings of Trefethen (Ex. 1006)—published by the same prestigious Cornell research team—to improve MATLAB’s parallel computing performance and extensibility.

Based on the above analysis, Petitioner’s showing of reasons to combine Menon and Trefethen is persuasive.

b. Motivation to Combine Menon (Ex. 1005) with RS6000 (Ex. 1007), MPIref (Ex. 1017), and POEref¹⁶ (Ex. 1008)

Regarding RS6000 (Ex. 1006), Petitioner contends that an ordinarily skilled artisan would have been motivated to “combine[] RS6000 with

¹⁶ POEref is part of Petitioner’s showing regarding claims 4 and 20 in ground 2 (i.e., not ground 1 and claim 1). However, Patent Owner presents parallel arguments regarding the motivation to combine RS6000, MPIref, and POEref, so the instant section analyzes Petitioner’s showing and Patent Owner’s arguments together.

Menon because Menon suggests the combination by teaching MultiMATLAB uses the ‘IBM SP2, a modern high performance distributed memory multiprocessor’ and RS6000 ‘provides ‘an overview of the IBM RS/6000 SP.’” Pet. 25–26 (citing Ex. 1003 ¶ 88; quoting Ex. 1005, 5; Ex. 1007, 1). In support of this contention, Petitioner argues that an ordinarily skilled artisan “seeking to understand MultiMATLAB would be motivated to seek documentation corresponding to the systems for which ‘MultiMATLAB was design[ed] specifically.” *Id.* at 26 (quoting Ex. 1005, 6). According to Petitioner, “Menon and RS6000 together represent a combination of prior art elements . . . according to known methods . . . to yield the predictable result of MultiMATLAB implemented on an IBM SP2 having the physical attributes disclosed by RS6000.” *Id.* (citing Ex. 1003 ¶¶ 159–162; Ex. 1005, Abstr., 6; Ex. 1007, 1, 11).

Petitioner also contends that an ordinarily skilled artisan would have had a reasonable expectation of success “because Menon expressly discloses implementing MultiMATLAB on an IBM SP2 and that MultiMATLAB ‘was design[ed] specifically for the IBM SP2.” Pet. 26–27 (citing Ex. 1003 ¶¶ 159–162; Ex. 1005, 6, 14).

Regarding MPIref (Ex. 1017), Petitioner contends that “Menon suggests the combination by expressly citing to ‘the MPI communication standard’” and “MPIref is the communication standard referenced by Menon.” Pet. 27 (quoting Ex. 1005, 5; citing Ex. 1017, 1). Petitioner next contends that “Menon indicates that a POSITA would look to use the MPI standard because of MPI’s ‘broad popularity, its suitability to high performance parallel computing, and the virtually universal availability of vendor-optimized versions on modern parallel computers.’” *Id.* (quoting Ex. 1005, 5). Petitioner also contends that “[t]he combination of MPIref with

Menon would have been obvious because Menon and MPIref together represent a combination of known prior art elements . . . to yield the predictable result of an implementation of MultiMATLAB utilizing the features of MPI disclosed by MPIref.” *Id.* at 28 (citing Ex. 1005, 3–10, Ex. 1017, ii). According to Petitioner, “there would have been a reasonable expectation of success at least because Menon expressly discusses using MPI, which is the subject of MPIref, to establish communication between all the MATLAB processes of MultiMATLAB.” *Id.* (citing Ex. 1005, 3; Ex. 1017, 2; Ex. 1003 ¶¶ 166–170).

Patent Owner argues that combining RS6000 (Ex. 1007) and POEref (Ex. 1008) with Menon (Ex. 1005) is “not a ‘combination of prior art elements according to known methods.’” Resp. 26. This naked assertion does not undermine the Petition. Patent Owner concedes that the Petition “assert[s] motivation to combine RS6000 with Menon because Menon uses the IBM SP2/RS6000 SP system.” *Id.* (citing Pet. 25–27). Patent Owner also concedes that the Petition “assert[s] motivation to combine POEref with Menon because Menon teaches MultiMATLAB uses POE.” *Id.* (citing Pet. 111–112).

The Petition shows persuasively that Menon (Ex. 1005) suggests the combination with RS6000 (Ex. 1007) by teaching MultiMATLAB uses the “IBM SP2, a modern high performance distributed memory multiprocessor” and RS6000 (Ex. 1007) “provides ‘an overview of the IBM RS/6000 SP.’” Pet. 25–26 (citing Ex. 1003 ¶ 88; quoting Ex. 1005, 5; Ex. 1007, 1). Therefore, an ordinarily skilled artisan “seeking to understand MultiMATLAB would be motivated to seek documentation corresponding to the systems for which ‘MultiMATLAB was design[ed] specifically.” *Id.* at 26 (quoting Ex. 1005, 6).

The Petition also shows persuasively that Menon (Ex. 1005) suggests the combination with POEref (Ex. 1008) by teaching that MultiMATLAB uses “POE, IBM's Parallel Operating Environment.” Pet. 111 (citing Ex. 1005, 6). As the Petition explains “[a] POSITA seeking to understand MultiMATLAB would be motivated to review the documentation corresponding to the systems expressly discussed in Menon.” *Id.* (citing Ex. 1005, 6). Further, “Menon describes using POE to initialize the MATLAB processes.” *Id.* (citing Ex. 1005, 6). And “POEref describe[s] that ‘the Parallel Operating Environment (POE)’ provides the command ‘poe’ ‘to load and execute programs on remote nodes’ and ‘initializ[e] the local environment.’” *Id.* at 111–112 (citing Ex. 1008, ix, 28).

Accordingly, Petitioner shows that combining teachings regarding the known systems, RS6000 (Ex. 1007), MPIref (Ex. 1017), and POEref (Ex. 1008), with Menon (Ex. 1005) and Trefethen (Ex. 1006), each designed and documented to function together as part of the MATLAB system, would have amounted to combining prior art elements according to known methods in a predictable fashion, obtaining features and benefits for running parallel routines as outlined above, contrary to Patent Owner’s arguments otherwise. Based on the above analysis, Petitioner’s showing is persuasive.

c. Teaching Away

“A reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant.” *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 567 F.3d 1314, 1327 (Fed. Cir. 2009) (quoting *Ricoh Co. v. Quanta Comput. Inc.*, 550 F.3d 1325, 1332 (Fed. Cir. 2008)).

Patent Owner contends that Menon (Ex. 1005) teaches away from the combination with Trefethen (Ex. 1006). Resp. 6. According to Patent Owner, “Menon’s critique of the first architecture and its explicit statement that Trefethen’s earlier system does not permit the relevant extensions makes it unreasonable to conclude that a POSITA would have been motivated to use the first generation architecture as a basis for achieving the claimed invention.” *Id.* at 34; *see also id.* at 31 (“Menon does not present Trefethen’s first-generation MultiMATLAB work as a stable platform to extend; it presents it as an architecture with meaningful limitations that required a different approach.” (citing Ex. 1005, 14)); *id.* at 32 (arguing that Menon explains Trefethen “could not take advantage of vendor-optimized MPI implementations” and “expressly states that Trefethen’s first-generation architecture as inadequate.”) (citing Ex. 1005, 14)).

Patent Owner next argues that “Menon’s most pointed teaching-away statement is directed to extensibility: the first MultiMATLAB architecture ‘does not permit extensions via parallel MEX routines.’” Resp. 33 (citing Ex. 1005, 14).

Petitioner replies that Menon (Ex. 1005) and Trefethen (Ex. 1006) “share two of the same authors” and both “describe aspects of the same system—MultiMATLAB.” Reply 13 (citing Pet. 23–24; Ex. 1005, 1; Ex. 1006, 1.). Petitioner argues that “Menon explicitly cites Trefethen and explains that Menon builds on the teachings of Trefethen.” *Id.* (citing Pet. 23; Ex. 1003 ¶ 81; Ex. 1005, 1–2; Ex. 1163 ¶ 38).

Petitioner next contends that “Menon describes *enhancing* the system of Trefethen,” “describes a list of ‘goals’ already achieved by MultiMATLAB and then states that the architecture has been ‘redesigned to achieve two **further** goals,’” and one of the goals “is ‘extensibility,’ which

Menon achieves by its description of ‘MEX routines’ to the system of Trefethen.” Reply 14 (citing Ex. 1005, 2–3; Ex. 1163 ¶ 41). According to Petitioner, “Menon’s addition of the ‘MEX routines’ to MultiMATLAB shows that such a modification was within the capabilities of a POSITA—in fact, the authors of the Menon paper *actually implemented* the modification.” *Id.* (citing Ex. 1005, 2–3; Ex. 1163 ¶¶ 40–44).

Petitioner’s showing reveals that Patent Owner’s teaching away argument is unavailing. First, Patent Owner’s teaching away argument is inconsistent with the strong evidence of a motivation to combine Menon (Ex. 1005) and Trefethen (Ex. 1006) discussed above. Second, Patent Owner’s teaching away argument is contrary to the evidence of record for the reasons given in Petitioner’s Reply and as outlined above and further below. *See* Reply 13 (e.g., “Menon explicitly cites Trefethen and explains that Menon builds on the teachings of Trefethen.”) (citing Pet. 23; Ex. 1003 ¶ 81; Ex. 1005, 1–3; Ex. 1163 ¶ 38).

Dr. Bajaj’s reply testimony further supports Petitioner:

Menon does not identify differences between the first “MultiMATLAB” architecture and the newer architecture, but any such differences do not teach away from using Trefethen’s command-level teachings. Ex. 1005, 2. For example, Menon describes a list of “goals” already achieved by MultiMATLAB and then states that the architecture has been “redesigned to achieve two **further** goals.” One of these two “further goals” is “extensibility.” Menon achieves this [extensibility by adding] “MEX routines” to the system of Trefethen. The “MEX routines” do not show an “incompatibility” between Menon and Trefethen. Instead, Menon’s addition of the “MEX routines” to MultiMATLAB shows that such a modification was within the capabilities of a POSITA because the authors of the Menon paper actually implemented the modification.

Ex. 1163 ¶ 41 (citing Ex. 1003 ¶ 71; Ex. 1005, 1–3). Dr. Bajaj’s credible testimony is unrebutted and supported by the record. Therefore, Petitioner shows that Menon enhances Menon’s system and does not teach away from the combination with Trefethen. *Id.* ¶ 41.

Specifically, as Petitioner argues and Dr. Bajaj testifies, Menon (Ex. 1005) does not “distinguish” the “MultiMATLAB” teachings of Trefethen. Reply 14; Ex. 1163 ¶ 14. Rather, Menon enhances Trefethen’s system. *See id.* (citing Ex. 1005, 2; Ex. 1163 ¶ 41). In particular, Menon lists “goals” of MultiMATLAB, and then states that the architecture has been “redesigned to achieve two **further** goals.” *Id.* (quoting Ex. 1005, 2, citing Ex. 1163 ¶ 41). One of these two “**further** goals” is extensibility, obtainable via the addition of “MEX routines” to the system of Trefethen. *Id.* (citing Ex. 1005, 2–3).

The record supports Petitioner as Menon (Ex. 1005) leads artisans of ordinary skill toward, and not away from, implementing the combined system. For example, Menon states that “[u]sers or other third parties may extend the environment with new high-performance parallel routines written in C or Fortran.” Ex. 1005, 2. Menon “discusses” “different methods” of “integration of high performance parallel routines into MATLAB” in Sections 4 and 5. *See id.* at 6. Menon states that “MATLAB’s MEX interface allows C and Fortran programs to be run directly from MATLAB.” And Menon further states that it “demonstrate[s] how parallel routines written in MPI can be integrated into MATLAB in our MPI-based Multi-MATLAB implementations,” wherein “[t]he general method is write or rewrite routines as parallel MEX routines.” *Id.* at 6.

And as Petitioner argues, there is no “‘incompatibility’ between Menon and Trefethen,” because “Menon’s addition of the ‘MEX routines’ to MultiMATLAB shows that such a modification was within the capabilities

of a POSITA—in fact, the authors of the Menon paper *actually implemented* the modification.” Reply 14 (citing Ex. 1163 ¶¶ 40–44; Ex. 1005).

As outlined above, the Petition provides additional reasons for combining the references, thereby leading artisans of ordinary skill toward the proposed combination of teachings: Menon (Ex. 1005) cites Trefethen and explains that Menon builds on Trefethen’s teachings. Pet. 24; Ex. 1003 ¶ 83; Reply 12. The combination yields predictable results according to known methods—MultiMATLAB system executing MATLAB in parallel to perform mathematical calculations by exercising MATLAB code on a multi-node system. Pet. 24; Ex. 1005, Abstract; Ex. 1006, 2–7; Ex. 1003 ¶ 82; Reply 12. The Petition shows that “Trefethen describes ‘how the [MultiMATLAB] system is used’ using code examples that perform parallel mathematical calculation using MATLAB” and “Menon describes ‘tak[ing] advantage of high performance distributed memory processors’ to enable MATLAB to perform mathematical calculations.” Pet. 24 (quoting Ex. 1005, Abstr.; Ex. 1006, 2; citing Ex. 1005, 16). The ability to perform mathematical calculations using the MATLAB code including MEX routines in the combined system is not insufficient “generic” motivation, contrary to Patent Owner’s arguments.

In other words, Menon’s description of building on and enhancing the teachings of Trefethen—the earlier work of the same research group at Cornell—by adding MEX routines to achieve high-performance parallel computing and extensibility for MATLAB, demonstrates a classic motivation for a POSITA to combine. As discussed above, Menon describes MPIref, RS6000, and POEref to further lead artisans of ordinary skill toward implementing operational details and standard interfacing techniques for the well-known parallel operating platform of Menon-Trefethen to teach a

POSITA how to implement the MultiMATLAB system with a reasonable expectation of success. Considering Menon and Trefethen in light of the knowledge of an artisan of ordinary skill reveals that the facts Petitioner relies upon undermine Patent Owner’s teaching away arguments.

d. Considering Motivation and Teaching Away Together

For the reasons outlined above, Petitioner’s evidence and contentions are persuasive to show sufficient reasons to combine the reference teachings without teaching away from Trefethen. Patent Owner’s unsupported attorney arguments do not undermine Petitioner’s showing. Petitioner establishes that an ordinarily skilled artisan would have been motivated to combine Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), MPIref (Ex. 1017), and POEref (Ex. 1008) with a reasonable expectation of success. *See Intel Corp. v. PACT XPP Schweiz AG*, 61 F.4th 1373, 1381 (Fed. Cir. 2023) (explaining that it is sufficient for motivation to combine “that there was a known problem . . . in the art, that Bauman[] . . . helped address that issue, and that combining the teachings of Kabemoto and Bauman wasn’t beyond the skill of an ordinary artisan. Nothing more is required to show a motivation to combine under *KSR*”).

d. Preamble 1.0: “A computer cluster comprising:”

Petitioner provides the following annotated version of Menon’s Figure 1:

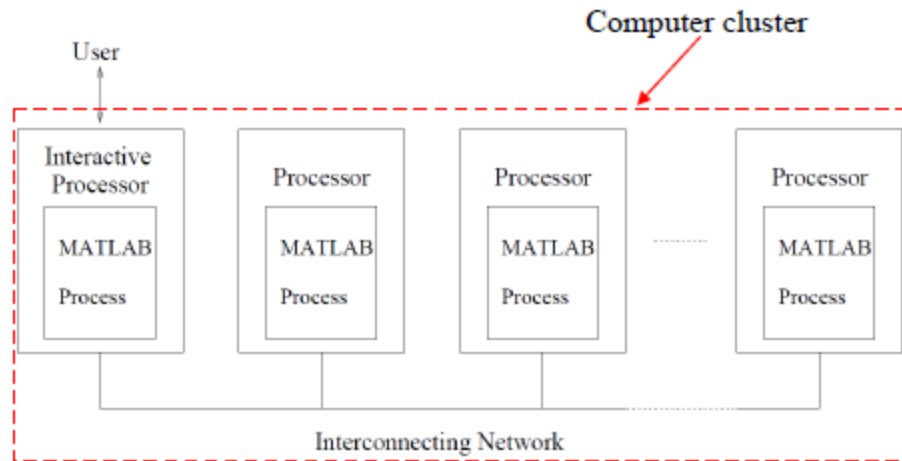


Figure 1: MultiMATLAB Architecture

Ex.1005, FIG. 1 (annotated); Ex.1003, ¶170

Pet. 29. Figure 1 is a block diagram illustrating “MultiMATLAB Architecture.” Ex. 1005, 3. Petitioner annotates Figure 1 with a broken line red rectangle surrounding four interconnected processors and text to indicate a “Computer cluster.” *Id.* at 29.

Petitioner relies on Menon’s disclosure that “MultiMATLAB was designed ‘specifically for the IBM SP2.’” Pet. 29 (quoting Ex. 1005, 3). Petitioner also relies on “RS6000’s description that the IBM SP2 [which Memon employs] is a networked cluster computing system for executing parallel programs.” *Id.* Also, Petitioner shows that Menon uses “a plurality of processor nodes to perform mathematical calculations.” *Id.* (citing Ex. 1005, 13, Figs. 7, 8 (using “8 processors” and “32 processors,” respectively)). Therefore, Petitioner asserts that the combined teachings render the preamble obvious (even if it is limiting). *See id.* (citing Ex. 1003 ¶¶ 171–177).

Patent Owner does not address the preamble. *See generally* Resp. Petitioner shows persuasively that the combination of Menon (Ex. 1005) and

RS/6000 (Ex. 1007) teaches the subject matter of the preamble.

Consequently, it is not necessary to determine if the preamble is limiting.

e. Limitations 1.1.1–1.1.3: “a first processor,” “a second processor,” and “a third processor”

Petitioner relies the combined teachings of Menon (Ex. 1005) and RS6000 (Ex. 1007) to address the three processors as recited in claim 1. Pet. 32–33.

Petitioner annotates Menon’s Figure 1 as follows (Pet. 32):

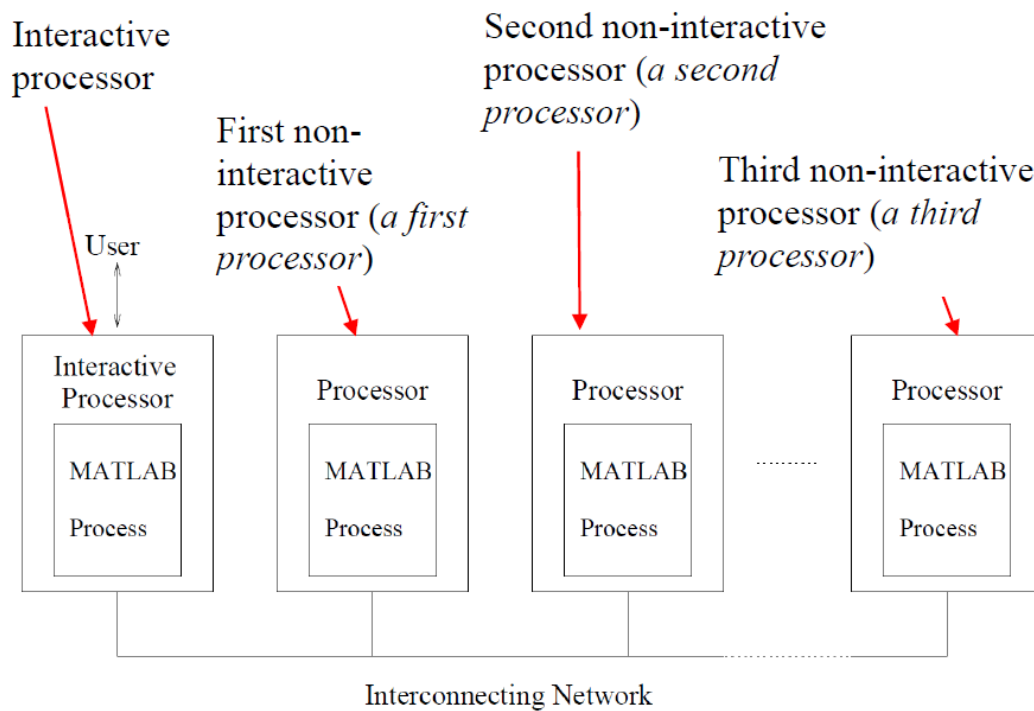


Figure 1: MultiMATLAB Architecture

Petitioner’s annotated version of Menon’s Figure 1 shows how Petitioner reads Menon’s MultiMATLAB architecture on the IBM SP2 as including the claimed first, second, and third processors (as annotated by Petitioner in italics). Pet. 32–33 (citing Ex. 1003 ¶ 180; Ex. 1005, 3 (“[E]ach processor in a parallel platform individually runs a MATLAB process”).

Patent Owner does not specifically address limitations 1.1.1, 1.1.2, or 1.1.3. *See generally* Resp. Petitioner shows persuasively that the combination of Menon and RS/6000 teaches limitation 1.1.

f. Limitation 1.2: “at least one computer-readable medium in communication at least one of the first processor, the second processor, or the third processor”

Petitioner relies on the combined teachings of Menon (Ex. 1005) and RS6000 (Ex. 1007) to address limitation 1.2. Pet. 33–34. Petitioner contends that each processor runs a MATLAB process with each including a MATLAB Process Address Space for communication with memory, and RS6000 teaches that “the IBM SP2 has disk drives and memory cards for storing program code accessible by the node’s hardware processors,” which “renders obvious” “at least one computer-readable medium” in communication with the recited processors. *See id.* at 33–34 (citing Ex. 1005, 3, Fig. 1; Ex. 1007, 6, 10; Ex. 1003 ¶¶ 186–190); *supra* § II.D.6.b (reasons for combining Menon and RS6000).

Patent Owner does not specifically address limitation 1.2. *See generally* Resp. Petitioner shows persuasively that the combination of Menon (Ex. 1005) and RS/6000 (Ex. 1007) teaches limitation 1.2.

g. Limitations 1.3.1, 1.5.1, and 1.7.1: “a first kernel residing in the at least one computer-readable medium,” “a second kernel residing in the at least one computer-readable medium,” and “a third kernel residing in the at least one computer-readable medium”

Petitioner contends that “Menon discloses the first non-interactive processor having MATLAB (*first kernel*) in an address space (*computer-readable medium*), rendering obvious this limitation.” Pet. 36 (citing Ex. 1003 ¶¶ 191–195, Fig. 1). Petitioner notes that “[t]he ’582 patent identifies MATLAB as an example of a *kernel*.” *Id.* (citing Ex. 1001, 1:16–17, 4:21–

26). For the second and third kernels of limitations 1.5.1 and 1.7.1, Petitioner relies on a similar showing and contends that each MATLAB process on a processor of the three processors in Menon's Figure 1 teaches *a kernel*. Pet. 45 (second kernel), 56 (third kernel).

Petitioner annotates Menon's Figure 1 to illustrate its showing, as follows (Pet. 35):

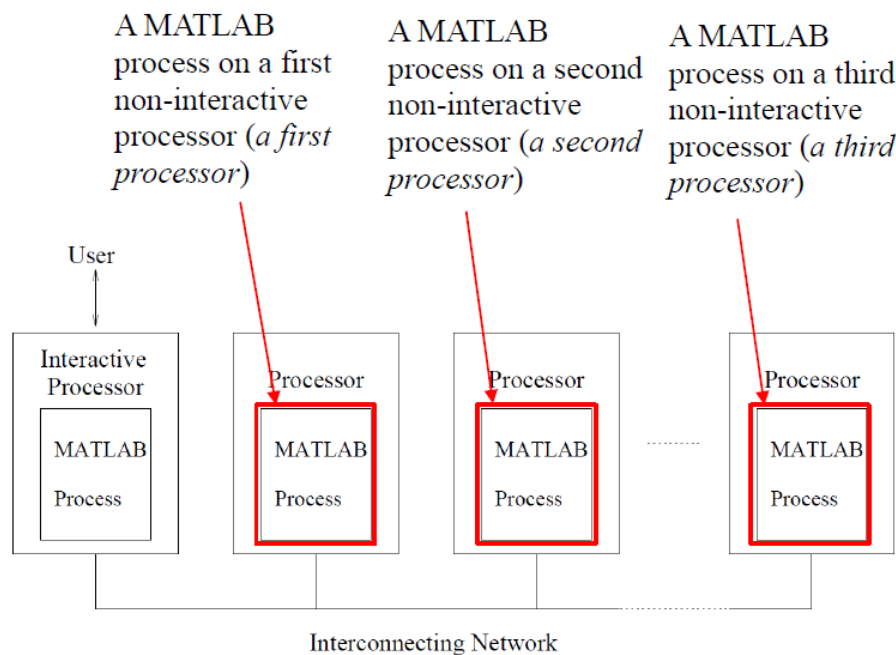


Figure 1: MultiMATLAB Architecture

Ex.1005, FIG. 1 (annotated); Ex.1003, ¶192

As annotated by Petitioner, Figure 1 above illustrates MATLAB as a process running in each of three non-interactive processors (red box, italics) in Menon's system. Pet. 35. Petitioner contends that "Menon explains that each MATLAB process of MultiMATLAB contains MATLAB, 'the numerical computing environment of choice on uniprocessors for hundreds of thousands of engineers and scientists.'" *Id.* (quoting Ex. 1005, 1; citing Fig. 2 (showing a MATLAB process)).

Petitioner also refers to its showing with respect to limitation 1.2 (based on the combined teachings of Menon and RS600) and asserts each MATLAB process in each processor is a kernel in “at least one computer-readable medium.” Pet. 34–35 (annotating Ex. 1005, Figs. 1, 2).

Patent Owner does not specifically address limitation 1.3.1, 1.5.1, or 1.7.1. *See generally* Resp. Petitioner shows persuasively that the combination of Menon (Ex. 1005) and RS/6000 (Ex. 1007) teaches limitations 1.3.1, 1.5.1, and 1.7.1.

h. Limitations 1.3.2 and 1.5.2: “said first kernel configured to translate user commands into calls for execution of matrix operations on an array of data elements,” and “said second kernel configured to execute of[sic] the matrix operations on the array of data elements and return a first result using the second processor”

Addressing limitation 1.3.2, Petitioner relies on its showing above that Menon teaches MATLAB as the first kernel. Pet. 37. Petitioner asserts that “Menon describes MATLAB as performing ‘high-level MATLAB operations’ such as ‘matrix multiplications[and] solve for eigenvalues,’ and provides example ‘MultiMATLAB implementations of the conjugate gradients algorithm’ that ‘consists of a matrix-vector multiplication at every iteration.’” *Id.* at 37 (quoting Ex. 1005, 10) (emphasis in original).

Petitioner also relies on Trefethen’s disclosure of “several examples of a MATLAB receiving commands that are translated into calls for execution.” Pet. 38–39 (citing Ex. 1006, 3, 4, 6 (describing calls executed by each MATLAB process)). Petitioner contends it would have been obvious to combine the teachings of Menon and Trefethen based on their similar use of MATLAB to perform mathematical operations using matrices in order to implement parallel processing with code according to

implementation details in Trefethen. *See id.* at 38–40; *supra* § II.D.6.a (reasons for combining Menon and Trefethen).

Addressing limitation 1.5.2, Petitioner relies on similar teachings in Menon (Ex. 1005) and Trefethen (Ex. 1006), contending that the limitations would have been obvious (for the reasons summarized above). *See* Pet. 46–47; *supra* § II.D.7.a (Motivation). Petitioner further relies on Trefethen’s “example algorithm (the ‘cycle.m’ m-file) where mathematical results are returned from one processor to the next.” Pet. 47 (citing Ex. 1006, 6).

Petitioner also asserts that

Menon teaching MATLAB in the MATLAB process of the second non-interactive processor, in view of Trefethen teaching an m-file where that MATLAB in the MATLAB process on the second non-interactive processor receives the variable “a=2,” doubles it to “a=4,” outputs the result “a=4,” and then sends it to the next MATLAB process, in further view of Menon and Trefethen teaching that each MATLAB process performs matrix-related mathematical evaluations on the array of data elements, renders obvious this limitation.

Pet. 47 (citing Ex. 1003 ¶¶ 223–228).

With the exception of presenting generic arguments directed to the limitation “second kernel configured to . . . return a first result” as recited in limitation 1.5.2, Patent Owner does not specifically address limitation 1.3.2 or 1.5.2. *See generally* Resp. Petitioner shows persuasively that the combination of Menon and RS6000 teaches the remaining limitations of 1.3.2 and 1.5.2. We address arguments regarding the “second kernel configured to . . . return a first result” below. *Infra* § II.D.7.o.

i. Limitation 1.4.1: “a first cluster node module residing in the at least one computer-readable medium,” and limitation 1.4.2, “said

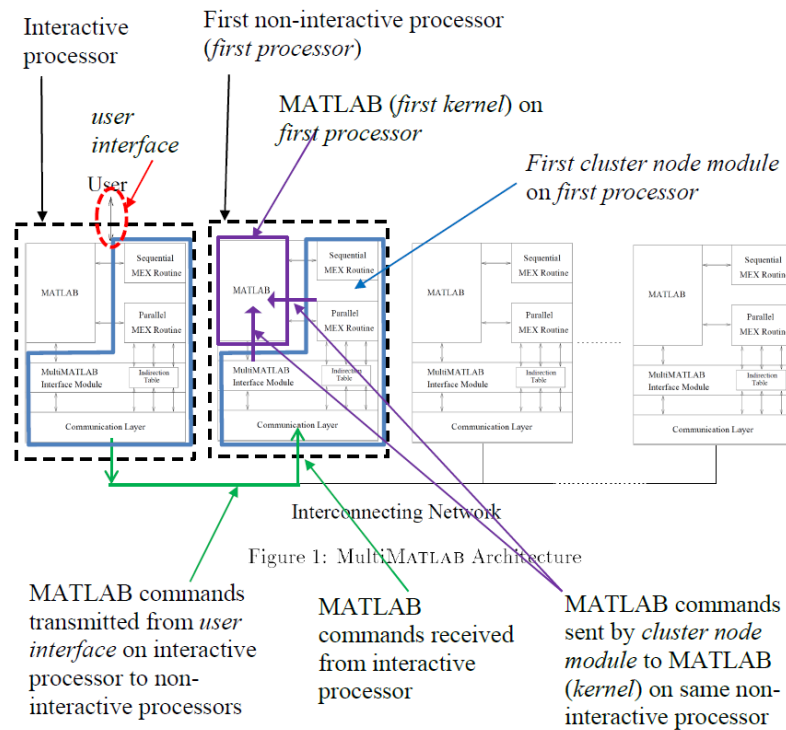
first cluster node module configured to communicate with the first kernel and receive commands from a user interface”

As summarized above in connection with limitation 1.2, Petitioner contends that the MATLAB process resides in an address space located in memory (computer-readable medium). Pet. 40. Petitioner adds that Menon’s “MATLAB process . . . includes a set of components to provide interconnection with other processes: ‘MEX Routines,’ ‘MultiMATLAB interface module,’ ‘indirection table’ and ‘communication layer’ (collectively, the *cluster node module*).” *Id.* (quoting Ex. 1005, 3, 5). Petitioner further contends that “‘MEX routines’ provide ‘[a]ll parallel functionality in the MultiMATLAB system.’” *Id.* (quoting Ex. 1005, 3). Petitioner also relies on Menon’s teaching that “[t]he interface module . . . is responsible for initializing an underlying communication layer, such as MPI . . . , and exposing it to the rest of the system.” *Id.* (quoting Ex. 1005, 3).

Patent Owner does not specifically address limitation 1.4.1. *See generally* Resp. Petitioner shows persuasively that the combination of Menon (Ex. 1005) and RS/6000 (Ex. 1007) teaches limitations 1.4.1.

j. Limitation 1.4.2: “said first cluster node module configured to communicate with the first kernel and receive commands from a user interface”

To address this limitation, Petitioner combines Menon’s Figures 1 and 2 and provides an annotated version thereof as follows (Pet. 44):



Ex.1005, FIGs. 1 and 2 (combined and annotated); Ex.1003, ¶217

As annotated above, Petitioner illustrates the first cluster node module (blue) on a first non-interactive processor (black dashed rectangle middle left) (with parallel and sequential MEX routines, MultiMATLAB Interface module, and communication module), in communication with a first (MATLAB) kernel (purple and purple arrows), receiving commands (green) from a first user interface (red) via Menon’s interactive processor (black dashed rectangle left). See Pet. 42–43 (citing Ex. 1005, 3–5).

Petitioner relies on Menon’s disclosure “that the ‘user interacts directly with one MATLAB process, called the interactive process, and operates within that process’s MATLAB environment’ to have user instructions executed in parallel by the MultiMATLAB architecture.” Pet. 43 (quoting Ex. 1005, 3; citing Ex. 1005, 3–4 (“On non-interactive processes, a separate top-level MEX routine . . . runs in a loop in which it

waits for MATLAB commands to arrive from the interactive process and executes them in its own MATLAB environment.”)).

Petitioner also relies on Trefethen’s similar disclosure of a user entering commands sent directly to multiple processors: “For example, ‘the user now types Eval(“sqrt(2)”),’ which causes ‘the MATLAB command sqrt(2) [to be] executed on all six processors.’” Pet. 44–45 (quoting Ex. 1006, 3). Petitioner also relies on Trefethen’s example of selecting a subset of processes by providing different process IDs. *Id.* at 44 (selecting processes ID=4 and ID=5).

Based on the above teachings and its stated reasons for combining the teachings (*supra* § II.D.6.a), Petitioner contends that Menon and Trefethen collectively teach the following sequence:

- (i) the user enters a command via the user interface on the interactive processor, (ii) that command is provided to the MEX routine of the cluster node module on the interactive processor, (iii) the cluster node module on the interactive processor communicates the command to the cluster node modules of the other processors, (iv) the cluster node module of the first processor receives the command via the interconnecting network, and (v) the cluster node module of the first processor communicates the command to the first kernel.

Pet. 45 (citing Ex. 1003 ¶¶ 211–219).

Patent Owner does not specifically address limitation 1.4.2 other than to generally discuss Petitioner’s showing regarding the “user interface” in a different context related to a “final result” as recited in limitation 1.9. *See generally* Resp. Petitioner otherwise shows persuasively that the combination of Menon (Ex. 1005) and Trefethen (Ex. 1006) teaches limitation 1.4.2. We address arguments related to the “user interface” below. *See infra* §§ II.D.7.n, o.

k. Limitations 1.6.1, 1.6.2, and 1.6.3: “a second cluster node module residing in the at least one computer-readable medium,” “said second cluster node module configured to send commands to the second kernel and” “communicates with the first cluster node module”

Petitioner relies on its similar showing for limitations 1.4.1 and 1.4.2 to address limitations 1.6.1, 1.6.2, and 1.6.3. *See* Pet. 48–55. Petitioner identifies a second cluster node module as residing on Menon’s second non-interactive processor. *See id.* at 55 (annotating Menon’s Figures 1 and 2 combined), 62 (same). Petitioner contends that “each MATLAB process is provided with ‘the ability to communicate with other processes through a communication layer that runs over the parallel platform’s interconnection network.’” *Id.* at 51 (quoting Ex. 1005, 3). Petitioner further contends that “it is the ‘MEX routines’ of the cluster node module that ‘access the underlying communication layer and, hence, the network, through the MultiMATLAB interface module.’” *Id.* (quoting Ex. 1005, 4).

According to Petitioner,

Menon and Trefethen teach that the MATLAB processes on the second non-interactive processor and the first non-interactive processor use their respective MEX routines, MultiMATLAB interface module, indirection table, and communication layer (*said second cluster node module* and *the first cluster node module*) to communicate commands (e.g., tasks) and data directly with each other, rendering obvious this limitation.

Pet. 55 (citing Ex. 1003 ¶¶ 234–243).

Patent Owner does not specifically address limitations 1.6.1, 1.6.2, or 1.6.3. *See generally* Resp. Petitioner shows persuasively that the combination of Menon (Ex. 1005) and Trefethen (Ex. 1006) teaches limitations 1.6.1, 1.6.2, and 1.6.3.

1. Limitation 1.7.2: “said third kernel configured to receive the first result and execute additional matrix operations on the first result and return a second result to one or more of the first, second, or third cluster node modules using the third processor”

Referring to its showing for limitations 1.1.3 and 1.5.2 (which addresses matrix operations), Petitioner contends that Trefethen’s “cycle.m” example “teaches that MATLAB in MATLAB process ID=3 doubles the received variable (‘a=4’) to ‘a=8’ (*a second result*) and sends it to the next MATLAB process, i.e., MATLAB process ID=4, as shown” in Trefethen’s process of doubling variables and sending successive results to different processors. *See* Pet. 56–57 (citing Ex. 1006; Ex. 1003 ¶ 249).

Petitioner also contends that

MATLAB (*the kernel*) communicates with other processes using its *cluster node module*; that is, for example, MATLAB process ID=3 on the third non-interactive processor sends the result “a=8” (*a second result*) to the next MATLAB process ID=4 via its *third cluster node module* (for the result to arrive at the next MATLAB process ID=4). A POSITA would recognize and understand that for *the third cluster node module* in MATLAB process ID=3 to send the result “a=8” (*a second result*) to the next MATLAB process, *the third cluster node module* receives the result “a=8” from MATLAB (*third kernel*).

Pet. 57 (citing Ex. 1005, Figs. 1, 2; Ex. 1003 ¶ 250).

Patent Owner does not directly address limitation 1.7.2 but generally presents arguments regarding whether a kernel receives a first result computed by another kernel and returns a second result, which we address below in Section II.D.7.o. *See* Resp. 11–16; *infra* § o.

m. Limitations 1.8.1, 1.8.2, and 1.8.3: “a third cluster node module residing in the at least one computer-readable medium,” “said third cluster node module configured to send commands to the third kernel and” “configured to communicate with the first cluster node module and the second cluster node module”

To address these limitations, Petitioner relies on its showings for limitation 1.4.1 (explaining “the MEX routines, MultiMATLAB interface module, indirection table, and communication layer (*cluster node module*) of each MATLAB process on a processor of MultiMATLAB are in the MATLAB process address space”), limitation 1.7.1 (explaining “MATLAB in the MATLAB process on the third non-interactive processor (third processor) teaches the third kernel”), 1.4.2 (explaining “the interactive processor sends the user commands to the cluster node module of each non-interactive processor, which provides MATLAB commands to its respective MATLAB (*kernel*) for execution”).

To illustrate its showing, Petitioner provides an annotated version of a combination of Menon’s Figures 1 and 2 as follows (Pet. 62):

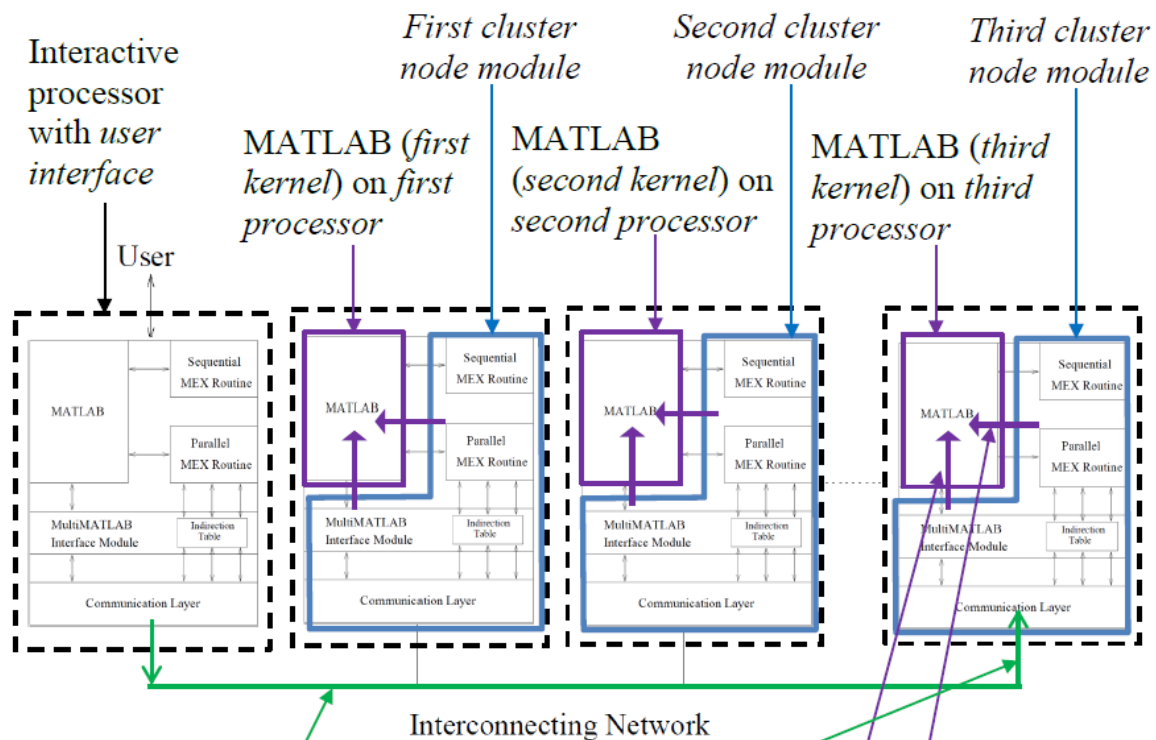


Figure 1: MultiMATLAB Architecture

MATLAB commands transmitted from *user interface* on interactive processor to non-interactive processors

MATLAB commands received from interactive processor

MATLAB commands sent by *cluster node module* to MATLAB (*kernel*) on same non-interactive processor

Ex.1005, FIGs. 1 and 2 (combined and annotated); Ex.1003 ¶258

Pet. 62 (citing Ex. 1003 ¶¶ 256–258). Petitioner’s figure above is a combination of Menon’s Figures 1 and 2 as annotated by Petitioner. *See* Ex. 1005, Figs. 1, 2. Petitioner’s annotated figure above illustrates three non-interactive processors (purple) of Menon’s four processors with MATLAB kernels connected directly to corresponding first, second, and third cluster node modules (blue), corresponding to claim 1’s three kernels and cluster node modules, all to the right of Menon’s interactive processor (black dotted line), which includes a user interface, corresponding to claim 1’s user

interface. Petitioner contends that as illustrated above in the combined Figures 1 and 2 of Menon,

the MEX routines (part of *said third cluster node module*) in the MATLAB process of the third non-interactive processor receive MATLAB commands from the interactive processor and send the MATLAB commands (*configured to send commands*) to the MATLAB (*to the third kernel*) in that MATLAB process for execution by that MATLAB.

Pet. 61 (citing Ex. 1003 ¶¶ 256–258).

Petitioner also contends that “[a]s discussed in [1.6.3], each MATLAB process has a cluster node module for communicating with the other cluster node modules. Further, each MATLAB process is configured to use the Eval(), Send(), and Recv() commands to send data from one process to another.” Pet. 62–63 (citing Ex. 1005, Table 1 (showing commands for point to point communication); Ex. 1003 ¶ 239). Petitioner further relies on Trefethen’s multiple examples of using the same MEX routines “to communicate results of mathematical expression evaluation to other MATLAB processes on other processors.” *Id.* at 63 (citing Ex. 1006, 6; Ex. 1005, 8–9, Table 1).

[Based on the above-summarized teachings, Petitioner contends that] Menon and Trefethen teach that the MATLAB processes on the third non-interactive processor, the second non-interactive processor, and the first non-interactive processor use their respective MEX routines, MultiMATLAB interface module, indirection table, and communication layer (*said third cluster node module, the second cluster node module and the first cluster node module*) to communicate commands (e.g., tasks) and data with each other, rendering obvious this limitation.

Pet. 64 (citing Ex. 1003 ¶¶ 259–261).

Patent Owner does not specifically address Petitioner’s showing regarding limitations 1.8.1, 1.8.2, or 1.8.3. Resp. 15. To the extent Patent

Owner's generic arguments regarding the process or communication flow implicit in claim 1, we address the arguments below. *See infra* §§ II.D.7.n, o. Petitioner shows persuasively that the combination of Menon and Trefethen otherwise teaches limitations 1.8.1, 1.8.2, and 1.8.3.

n. Limitation 1.9: “wherein the first cluster node module is configured to return, based on the second result, a final result.”

Referring to its showing with respect to limitation 1.7.2, Petitioner contends “Trefethen teaches that, when executing ‘the m-file cycle.m,’ the final MATLAB process is programmed to output the results to the screen of the interactive process.” Pet. 65 (citing Ex. 1006, 6; Ex. 1003 ¶ 263). Petitioner also contends that “[t]o use MultiMATLAB in the complicated mathematical algorithms contemplated by Menon and Trefethen, Trefethen describes that ‘any of the MATLAB processes’ can perform ‘general point-to-point communication’ using ‘send and receive commands.’” *Id.* at 66 (quoting Ex. 1006, 5). Relying on its showing for limitation 1.4.2, Petitioner also asserts that “the interactive processor has a user interface for providing intermediary and final results of the mathematical calculations to the user.” *Id.* at 67.

Petitioner relies on other MATLAB teachings in Menon (Ex. 1005) and Trefethen (Ex. 1005) and contends as follows:

[B]ecause Menon and Trefethen teach that each MATLAB process (e.g., the first non-interactive process) is configured to receive a result from any other MATLAB process via its cluster node module, and to perform subsequent processing and returning the final result to the interactive processor, Menon and Trefethen render obvious this limitation.

Pet. 67 (citing Ex. 1003 ¶¶ 262–270).

Patent Owner responds that “[f]orwarding screen output to the master console is not the same thing as returning a computational result as a data

object that can be consumed by another kernel or used to generate a final result.” Resp. 12. Patent Owner contends that “Trefethen’s own description makes clear that result values are not implicitly returned through Eval(); they remain local to the executing process unless explicitly transmitted through separate messaging operation.” *Id.* at 12–13. Patent Owner also argues that “Menon similarly describes Eval() as a mechanism for distributing commands to MATLAB processes over an underlying communication layer[] but does not describe any mechanism by which the results of those commands are programmatically returned.” *Id.* at 13 (citing (Ex. 1005, 3–6). According to Patent Owner, “[s]creen output is not a returned result,” because “[s]creen output is a display artifact—it may show a value to a user, but it does not constitute a programmatic return of a result to another system component for further computation.” *Id.* at 13–14. According further to Patent Owner, “[i]f Eval() returned computational results, there would be no need for Put, Get, Send, or Recv. The fact that those commands exist—and are described as the mechanisms for moving data—confirms that result return is not implicit.” *Id.* at 14.

Patent Owner acknowledges that “Dr. Bajaj asserts that a person of ordinary skill would understand MultiMATLAB to return results of computations executed via Eval().” *Id.* However, Patent Owner contends that “those assertions are unsupported by the cited references and directly contradict the implementation descriptions in Trefethen and Menon.” *Id.* In summary, Patent Owner argues that “Petitioner cannot establish that the asserted prior art discloses programmatic return of evaluation results.” *Id.* at 15.

As Petitioner explains in its Reply, “Patent Owner acknowledges that the ‘screen output’ in Menon/Trefethen ‘**show[s] a value to a user,**’ in

particular a final value based on the value of the result communicated from the third kernel and received by the first cluster node module of the first non-interactive processor.” Reply 8–9 (citing Response 13). There is no claim distinction between showing a value to a user at the user interface and returning a result to the user interface, because “[t]hose values are data.” See Ex. 1163 ¶ 26 (testifying that Trefethen expressly states that MultiMATLAB uses “Eval, Send[and] Recv” to “pass commands and **data** between . . . various [MATLAB processes]” (quoting Ex. 1006, Abs.)). As Patent Owner acknowledges, Menon-Trefethen discloses “values . . . printed to a screen.” Resp. 14.

As discussed above (§ 1), limitation 1.7.2 recites “said third kernel configured to . . . return a second result to one or more of the first, second, or third cluster node modules using the third processor” and limitation 1.9 recites “the first cluster node module is configured to return, based on the second result, a final result.” See Reply 8.

As Petitioner explains,

Menon/Trefethen teaches that the first cluster node module of the first non-interactive processor receives the result from the third kernel, performs subsequent processing to the received result to generate a final result, and then sends that to the interactive processor, where *the user interface of the interactive processor displays it*.

Reply 9 (citing Ex. 1006, 6–7; Pet 65–67; Ex. 1003 ¶¶ 262–270; Ex. 1163 ¶¶ 20–21, 23–24, 26–27, 29; Reply § II.A.1).

Dr. Bajaj’s un rebutted testimony supports Petitioner:

Trefethen discloses that a user “connected to a node of the IBM SP2, running MATLAB,” in a MultiMATLAB architecture that includes additional MATLAB processes, *can execute on all the MATLAB processes using Eval, “[t]he standard MultiMATLAB command for executing commands on one or more processors.”*

That is, “[e]ach command passed to Eval was executed on all MATLAB processes.” The outputs from the MATLAB processes are “sent to the master process as soon as it is ready,” i.e., *without order or synchronicity between the MATLAB processes*. In the context of SPMD point-to-point communication, a “master process” is a “pseudo-master [that] is arbitrarily chosen to be the task [that] all the peers know a priori who to send the results back to.” Ex. 1011, 64.

Trefethen also *discloses peer-to-peer communication*, in that “[m]ore general point-to-point communication is *accomplished by send and receive commands*, which can be executed on any of the MATLAB processes.” For example, “SPMD programs can be built upon Send and Recv commands.”

Ex. 1003 ¶¶ 126–127 (citing Ex. 1006, 3–5; Ex. 1011, 23) (emphases added).

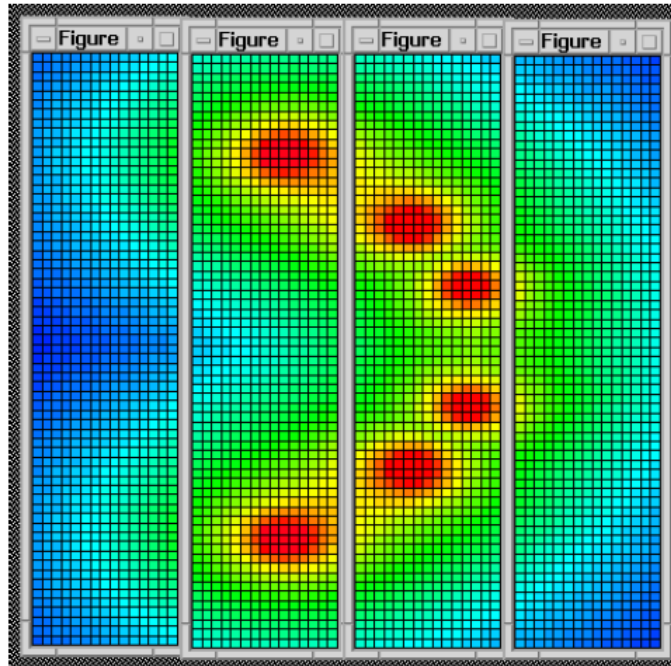
Dr. Bajaj supports Petitioner’s showing and further testifies as follows:

Menon and Trefethen also teach that each MATLAB process (e.g., the first non-interactive process) is configured to receive a result from any other MATLAB process via its cluster node module, to perform subsequent processing and returning the final result to the interactive processor [and] Menon teaches an algorithm (the conjugate gradient algorithm) in which each kernel is processing in parallel, and awaiting results from the other processes to interactively calculate further results, which includes the third process using the results from the second process and the first process using the results from the third process, and ultimately displaying the final result to the user via the user interface.

Ex. 1163 ¶ 17 (citing Ex. 1005, 11–14, Ex. 1011, 64; Ex. 1003 ¶ 267).

In short, Trefethen (Ex. 1006) discloses and teaches a mechanism for communicating a result of a mathematical computation at each node directly to the pseudo-master interactive process node (“*first node*”) in a peer-to-peer architecture, e.g. by executing the example script (m-file) “cycle.m” and SPMD program commands on MultiMATLAB.

Trefethen provides an example of an exemplary display:



Ex. 1006, 7–8. In the above figure, Trefethen illustrates a graphic example of a “4 by 1 grid of windows” that displays computational output—“computed . . . pseudospectra” of a “Grcar matrix”—to a user.

Based on the above summary and analysis, including respective positions of the parties, Dr. Bajaj’s testimony as quoted above is persuasive and consistent with the disclosures in Menon-Trefethen.

During the oral hearing, Patent Owner agreed that its arguments regarding a “final result” turn on its premise that “screen output is not data.” *See* Tr. 46:2–13, Tr. 53:4–20 (Q: “So your whole distinction is that screen output is not data?” A: “Correct.”), 77:19–21. However, as Petitioner shows as summarized above, Menon-Trefethen discloses the display of data values to a user. *See, e.g.*, Ex. 1163 ¶ 26; Ex. 1006, Abs. (“data” passes between “processes”). At another point during the oral hearing, Patent Owner further attempted to distinguish Menon-Trefethen by arguing that Menon-Trefethen’s display “can’t do anything with the data,” yet with

respect to claim 1, “the master node . . . can do something with the data.” *Id.* at 46:11–13. This argument is beyond the scope of claim 1 because claim 1 does not require any processing of the final result (and it is not clear what a “master node” is in claim 1). Therefore, the record supports Petitioner’s showing.

Based on the foregoing discussion, Petitioner shows persuasively that the combination of Menon and Trefethen teaches limitation 1.9.

o. Analysis of Patent Owner’s Remaining Arguments and
Petitioner’s Showing

Patent Owner generally argues that “Petitioner does not identify any disclosure in Menon or Trefethen in which a kernel receives a result computed by another kernel and performs additional matrix operations on that received result.” Resp. 16. Patent Owner also argues that “[i]f Eval() returned computational results, there would be no need for Put, Get, Send, or Recv. The fact that those commands exist—and are described as the mechanisms for moving data—confirms that result return is not implicit.” *Id.* at 14.

Patent Owner further generally argues that the proposed combination would not yield the claimed system. Resp. 27. According to Patent Owner, the control model of the asserted reference “is not the claimed execution model” because “Menon explicitly describes an architecture in which one process is ‘considered the interactive MATLAB’ and ‘all other processes wait for commands from the interactive MATLAB process,’ i.e., a master-controlled dispatch model.” *Id.* at 28 (quoting Ex. 1005, 6). Patent Owner contrasts this with what it argues is “the multi-stage, dependent evaluation and result-return architecture required by claim 1, which requires a second kernel to return a first result, a third kernel to receive that first result and

return a second result, and a first cluster node module to return a final result based on the second result.” *Id.*

These arguments are unavailing. The arguments do not meaningfully address, much less undermine, Petitioner’s showing regarding the flow of data through the kernels and cluster node module control as identified in Menon-Trefethen as summarized above, which shows, contrary to Patent Owner’s arguments, a second kernel configured to return a first result (limitation 1.5.1), a third kernel configured to receive that first result and return a second result (limitation 1.7.2), and a first cluster node configured module to return a final result based on the second result (limitation 1.9). *See supra* §§ II.D.7, g, l, n. In addition, the Petition does not rely solely on the Eval command, Patent Owner’s arguments confirm that “commands exist” “for moving data” in Menon-Trefethen’s system, and Petitioner relies on such commands. *See* Resp. 28; *accord* Pet. 62–63 (“[T]he Eval(), Send(), and Recv() commands . . . send data from one process to another.” (citing Ex. 1005, Table 1 (showing commands for point to point communication); Ex. 1003 ¶ 239)).

The whole purpose of MATLAB as described by Menon (Ex. 1005) and Trefethen (Ex. 1006) according to Petitioner’s showing is to return results based on parallel processing, in the form of matrix values, eigenvalues, etc. *See* Ex. 1005, 2 (discussing Trefethen as a general MATLAB extension), 5 (discussing MATLAB’s Differential Equation Tool Kit), 8 (describing retrieving results including eigenvalues from the slave processors using e.g., Put, Eval, and Get commands). As Petitioner also shows, “Menon and Trefethen both teach that the calculations performed at each kernel can be matrix operations” and each kernel depends on the output of a previous kernel. *See* Reply 7 (citing Ex. 1006, 4–5; Ex. 1005, 10; Pet.

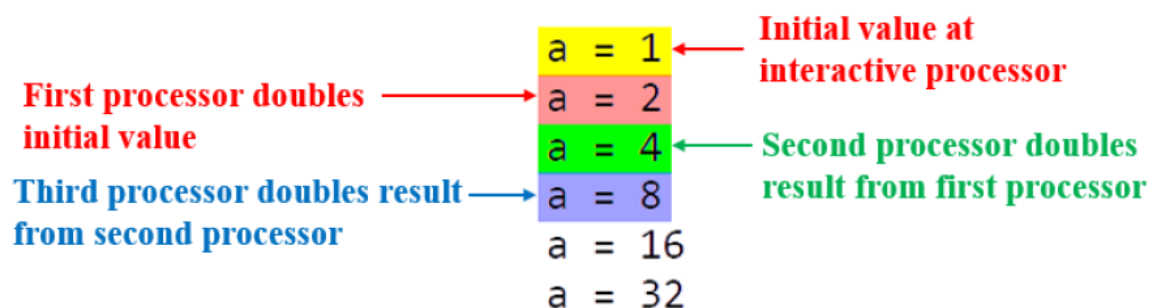
37–39); Ex. 1163 ¶ 19 (describing how each kernel in Trefethen doubles the value of a and sends it to another kernel (citing Ex. 1006, 6)); Ex. 1006, 6 (“Process 1 receives the message, doubles the value of a , and sends it along to process 2; and so on. If there are six processors the command Eval(‘cycle’) produces the output $a = 1$ $a = 2$ $a = 4$ $a = 8$ $a = 16$ $a = 32$.”).

The Petition and Reply both explain in detail how claim 1 reads on the combined teachings:

For example, in its description of the “cycle.m” script, Trefethen discloses that “[p]rocess 0” “creates the variable a with value 1 and sends it to process 1” (the *first kernel* on the *first processor*). Ex.1006, 6; Ex.1003, ¶¶127, 199; Petition, 21, 38–39. “Process 1” (the *first kernel* on the *first processor*) “**receives the message, doubles the value of a , and sends it along to process 2**” (the second kernel on the second processor) “and so on.” Ex.1006, 6; Ex.1003, ¶225; Petition, 47. And as described above, Menon and Trefethen both teach that the calculations performed at each kernel can be matrix operations. Ex.1006, 4–5; Ex.1005, 10; Petition, 37–39.

Reply 7; *accord* Pet. 56–60. Petitioner further explains as follows:

Trefethen further describes that each MATLAB process transmits its output “to the master process” (i.e., the cluster node module on the interactive processor) “as soon as it is ready.” Ex.1006, 4; Petition, 95; Ex.1003, ¶¶126, 349. When this sequence is executed in an example MULTIMATLAB system including six nodes, the user interface of the interactive processor produces the following output:



See Reply 4–5. In figure above, which Petitioner annotates and reproduces from Trefethen (Ex. 1006), Petitioner shows the results of each processor kernel successively doubling a result from a previous kernel with the initial value input via the user interface to the first interactive processor. *See* Reply 4–5 (reproducing Ex. 1006, 6 (annotated); citing Ex. 1163 ¶ 12; Pet 47, 57–58; Ex. 1003 ¶¶ 225, 249–250). As Petitioner explains, “[t]he value of a from the second non-interactive processor (4) is doubled again by the third non-interactive processor to 8 (“ $a=8$ ” in fourth line).” Reply 5 (citing Ex. 1006, 6; Pet. 56–58; Ex. 1003 ¶¶ 247–251). Finally, “that value of a from the third non-interactive processor (“ $a=8$ ”) is sent to the interactive processor and displayed to the user.” *Id.* (citing Ex. 1006, 6–7; Pet. 56–58; Ex. 1003 ¶¶ 247–251; Ex. 1163 ¶¶ 13–14).

To illustrate its showing for limitation 1.7.2, Petitioner provides an annotated version of a combination of Menon's Figures 1 and 2 as follows (Pet. 58):

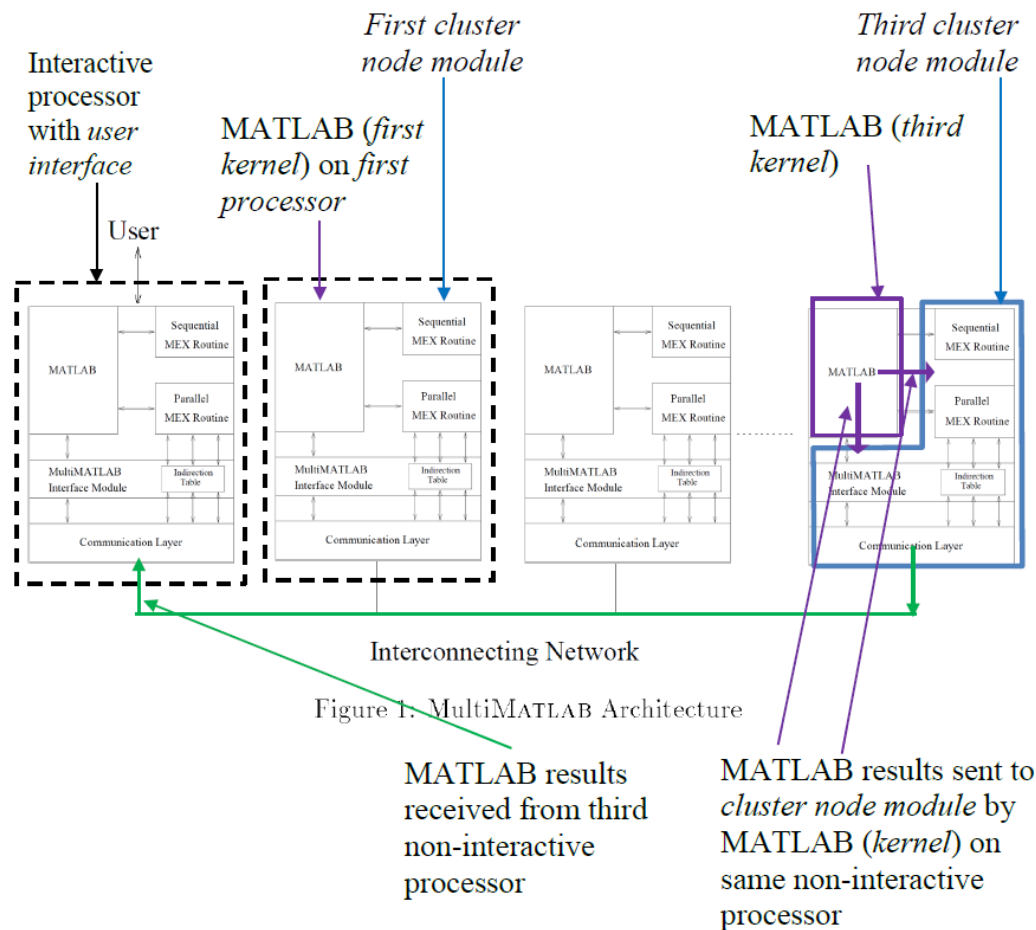


Figure 1. MultiMATLAB Architecture

Ex.1005, FIGs. 1 and 2 (combined and annotated); Ex.1003, ¶250

As illustrated above, combined Figures 1 and 2 from Menon as annotated by Petitioner illustrate the third kernel configured to transmit results to the third cluster node module and from there to any other cluster node module and ultimately to the User at the first interactive processor. *See* Pet. 58–59 (citing Ex. 1005, Figs. 1–2; Ex. 1003 ¶ 250).

Dr. Bajaj credibly supports Petitioner's showing with citations to the Petition, Trefethen (Ex. 1006), and Menon (Ex. 1005). *See* Ex. 1163 ¶¶ 13–

17. For example, regarding Patent Owner’s argument that Petitioner fails to identify wherein a “kernel receives a result computed by another kernel and performs additional matrix operations on that received result” (Resp. 16), Dr. Bajaj testifies as follows:

Menon and Trefethen also teach that each MATLAB process (e.g., the first non-interactive process) is configured to receive a result from any other MATLAB process via its cluster node module, and to perform subsequent processing and returning the final result to the interactive processor For example, Menon teaches an algorithm (the conjugate gradient algorithm) in which each kernel is processing in parallel, and awaiting results from the other processes to interactively calculate further results, which includes the third process using the results from the second process and the first process using the results from the third process, and ultimately displaying the final result to the user via the user interface.

Ex. 1163 ¶ 17 (citing Ex. 1005, 11–14, Ex. 1011, 64; Ex. 1003 ¶ 267). Dr. Bajaj further notes that Patent Owner’s argument “overlooks the dependent data-flow taught by Trefethen’s cycle.m example and the matrix-processing capability taught by both Menon and Trefethen,” wherein “Menon and Trefethen both teach that calculations performed at each kernel can be matrix operations.” *Id.* ¶¶ 18–19 (citing Ex. 1006, 6; Ex. 1003 ¶¶ 127, 199, 225).

After analyzing the relevant portions of Trefethen and Menon (Ex. 1163 ¶¶ 10–23), Dr. Bajaj testifies that “the first cluster node module of the first non-interactive processor is configured to receive a result from the third kernel/third cluster node module, perform or coordinate subsequent processing based on the received result as needed, and return a final result to the interactive processor, where *the user interface displays it*. Ex. 1163 ¶ 24 (citing Ex. 1006, 6–7; Ex. 1003 ¶¶ 262–270). Menon (Ex. 1005) and

Trefethen (Ex. 1006) support Dr. Bajaj’s cited testimony, and it is un rebutted.

Petitioner also shows that “[a]s discussed in [1.6.3], each MATLAB process has a cluster node module for communicating with the other cluster node modules. Further, each MATLAB process is configured to use the Eval(), Send(), and Recv() commands to send data from one process to another.” Pet. 62–63 (citing Ex. 1005, Table 1 (showing commands for point to point communication); Ex. 1003 ¶ 239). Petitioner further relies on Trefethen’s multiple examples of using the same MEX routines “to communicate results of mathematical expression evaluation *to other MATLAB processes on other processors.*” *Id.* at 63 (citing Ex. 1006, 6; Ex. 1005, 8–9, Table 1) (emphasis added). In other words, Trefethen describes that “any of the MATLAB processes” can perform “general point-to-point communication” using “send and receive commands.” *Id.* at 66 (quoting Ex. 1006, 5).

[Based on the above-summarized teachings, Petitioner contends that] Menon and Trefethen teach that the MATLAB processes on the third non-interactive processor, the second non-interactive processor, and the first non-interactive processor use their respective MEX routines, MultiMATLAB interface module, indirection table, and communication layer (*said third cluster node module, the second cluster node module and the first cluster node module*) to communicate commands (e.g., tasks) and data with each other, rendering obvious this limitation.

Pet. 64 (citing Ex. 1003 ¶¶ 259–261). Menon (Ex. 1005), Trefethen (Ex. 1006), and Dr. Bajaj (Ex. 1003) support this showing at the citations provided in the Petition.

In summary, Patent Owner’s arguments do not undermine, much less address with particularity, Petitioner’s specific showing wherein each kernel

is part of the MultiMATLAB process, and kernels successively process results from a prior kernel until a final kernel passes the desired result via cluster node modules (which communicate with each other via the interconnecting network annotated in green above) to the user interface—the whole purpose of MATLAB—which the record supports per specific citations by Petitioner. *See* Pet. 28–67; Ex. 1163 ¶¶ 10–19.

Based on the foregoing discussion, Petitioner shows persuasively that claim 1 would have been obvious.

m. Summary of Claim 1

The final record supports Petitioner’s showing for claim 1 as summarized above. We determine that Petitioner shows by a preponderance of evidence that claim would have been obvious.

8. Analysis of Claims 2, 3, and 5–19

Petitioner contends that claims 2, 3, and 5–19 would have been obvious over the combined teachings of Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), and MPIref (Ex. 1017). Petitioner presents contentions and evidence concerning why an ordinarily skilled artisan would have been motivated to combine Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), and MPIref (Ex. 1017) with a reasonable expectation of success. Pet. 68–110. Petitioner also maps the prior art to the limitations of claims 2, 3, and 15–19. *Id.* Patent Owner does not address Petitioner’s showing with respect to claims 2, 3, and 15–19 in its Response. *See generally* Resp.

Claim 14 recites “[t]he computer cluster of claim 1, wherein the first cluster node module is configured to return the final result to the user interface.” As set forth above, for limitations 1.4.2 and 1.9, Petitioner persuasively relies on the combined teaching of Menon-Trefethen to teach a

user interface. Petitioner further shows that Menon’s first cluster node module is configured to return the final result to the user “as soon as it is ready.” Pet. 95–96 (citing Ex. 1005, 3 (“The user interacts directly with one MATLAB process, called the interactive process, and operates within that process’s MATLAB environment.”), 5; quoting Ex. 1006, 4).

Regarding claim 14, Patent Owner argues in its Sur-reply that “[d]isplay to a user may follow a return to the user-interface, but the display event itself is not the return required by claim 14.” Sur-reply 5. This argument is improper because the Response does not set forth an argument for claim 14, so the argument raises a new theory not responsive to the Reply. *See* 37 C.F.R. § 42.23(b) (“A sur-reply may only respond to arguments raised in the corresponding reply”); *Rembrandt Diagnostics*, 76 F.4th at 1383 (“We have held that in some circumstances the Board acts within its discretion when declining to consider . . . a new theory of patentability raised by patent owner in sur-reply.”); *ParkerVision, Inc. v. Vidal*, 88 F.4th 969, 980–81 (Fed. Cir. 2023) (holding the Board did not abuse its discretion in excluding ParkerVision’s sur-reply arguments “because they proceeded in a ‘new direction’ relative to ParkerVision’s patent owner’s response”).

Even considering this Sur-reply argument, it is unavailing for similar reasons as discussed with respect to claim 1’s recited return of a final result in limitation 1.9, which Petitioner shows is configured to return to a user-interface. *See supra* § II.D.7.n; Pet. 95–96. In addition, as Patent Owner notes, Dr. Bajaj testifies that “a returned result may be returned to a user interface” because “[a] user interface is the component that presents information to the user.” *See* Sur-reply 4 (quoting Ex. 1163 ¶ 29; *supra* § n; Ex. 1163 ¶ 26 (testifying that Trefethen expressly states that MultiMATLAB

uses “Eval, Send [and Recv” to “pass commands and data between . . . various [MATLAB processes]” (quoting Ex. 1006, Abs.)).

Accordingly, Petitioner persuasively argues that claim 14 would have been obvious.

Claim 15 recites “[t]he computer cluster of claim 1, further comprising a communications network interface for the cluster node modules to communicate, with each other using a peer-to-peer architecture, results of execution of the matrix operations.” Building on its showing for claim 1, Petitioner shows that Menon (Ex. 1005) discloses the following:

Menon’s MEX routines use a “point to point . . . communication” paradigm. Ex.1005, 9. Unlike a “master/slave paradigm,” Menon’s SPMD/Message Passing paradigm uses “point to point . . . communication” to provide a “finer degree of communication.” Ex.1005, 9. Table 1 of Menon lists point-to-point commands, including “Send(pid,data),” which “send[s] data from one process to another,” and “Recv(pid),” which “receive[s] data sent from another process.” Ex.1005, Table 1.

Pet. 99. Based on this disclosure, Petitioner states that “Menon’s SPMD point-to-point communication paradigm where each node can communicate tasks and data directly with each other node renders obvious *a peer-to-peer architecture.*” *Id.* at 100. Petitioner also contends that Trefethen discloses MultiMATLAB commands and sending tasks and data to other processor nodes. *Id.* at 98–99 (citing Ex. 1006, 4, 6, 10). Petitioner also relies on RS6000 (Ex. 1007) and states as follows:

Menon’s MEX routines that enable each processor node to communicate tasks and data directly with the other nodes over an interconnecting network implemented on an IBM SP2, in view of Trefethen’s teaching that MEX routines enable communicating results of matrix operations from any process to another process, further in view of RS6000 teaching that each IBM SP2 node uses Ethernet and other communication adapters

to communicate over a message passing network, renders obvious this limitation. Ex.1003, ¶¶353-359.

Pet. 101.

Patent Owner argues that “[t]he Petition’s claim 15 analysis distinguishes Menon’s ‘point to point . . . communication’ paradigm from a ‘master/slave paradigm’ — the very architecture the construction excludes — and relies on Menon’s distinction to bridge the gap to peer-to-peer.” Sur-reply 1. Pet. 99 (citing Ex. 1005, 9). According to Patent Owner, “under Petitioner’s adopted construction, an architecture in which results pass through a central master process does not satisfy claim 15,” and “Trefethen’s MultiMATLAB is precisely that excluded architecture.” *Id.* at 2.

For similar reasons to that for claim 14, this argument is untimely and improper because the Response does not set forth an argument for claim 15, so the argument raises an entirely new theory not responsive to the Reply.

Even if we consider the arguments, the combination does not violate Petitioner’s construction, which follows: A “peer-to-peer architecture” is one “in which each node can communicate tasks and data with other nodes without the tasks and data being required to go through a central server or master node.” *See* Sur-reply 1; Pet. 15. Even if a final output eventually passes through Menon-Trefethen’s interactive processor, according to Petitioner’s showing and as summarized above, Menon-Trefethen’s outputs also pass from node (noninteractive processor) to node (another noninteractive processor) wherein each node doubles the value of “*a*” without that doubled value required to pass through the interactive processor. *See supra* § II.D.7.o (discussing Reply 4–5 (reproducing Ex. 1006, 6 (annotated); citing Ex. 1163 ¶ 12; Pet 47, 57–58; Ex. 1003 ¶¶ 225, 249–250)). For example, as Petitioner explains, “[t]he value of *a* from the

second non-interactive processor (4) is doubled again by the third non-interactive processor to 8 (“a=8” in fourth line.” Reply 5 (citing Ex. 1006, 6; Pet. 56–58; Ex. 1003 ¶¶ 247–251); *accord* Pet. 56–59 (showing the doubling of values from kernel to kernel without passing all such results to the display).

In particular, Petitioner shows that “Trefethen describes that ‘any of the MATLAB processes’ can perform ‘general point-to-point communication’ using ‘send and receive commands.’” Pet. 66 (quoting Ex. 1006, 5). That is, “[t]he Send(), Eval(), and Recv() MEX routines (built on MPI message passing) send results directly from one MATLAB process to another process.” *Id.* (citing Ex. 1005, 3; Ex. 1006, 5–6). “For example, the Send(), Eval(), and Recv() MEX routines can send a result to any other process by simply specifying the process ID.” *Id.* (citing Ex. 1006, 3, 5–6). In other words, Petitioner shows that Menon-Trefethen’s point-to-point communication architecture does not require passing intermediate results through a “central master,” contrary to Patent Owner’s arguments. *See* Pet. 65 (MEX routines can send a result to any other process by simply specifying the process ID” (citing Ex. 1006, 3, 5–6)) and peers can employ point to point communication and send results to a “pseudo-master” (citing Ex. 1011, 64)).

Accordingly, Petitioner persuasively shows how claims 1 and 15 read on Menon-Trefethen’s modified system without violating Petitioner’s claim construction.

Based on the foregoing discussion and a review of the record, Petitioner shows by a preponderance of evidence that claims 2, 3, and 5–19 would have been obvious.

E. Ground 2: Alleged Obviousness of claims 4 and 20

Claim 4 recites “[t]he computer cluster of claim 1, further comprising a cluster configuration module that initializes at least one of the first cluster node module, the second cluster node module, or the third cluster node module.”

Claim 20 recites “[t]he computer cluster of claim 1, wherein one or more of the cluster node modules are configured to receive a command to start a cluster initialization process for the computer cluster.”

Petitioner contends that claim 4 and 20 would have been obvious over the combined teachings of Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), MPIref (Ex. 1017), and POEref (Ex. 1008). Pet. 110–114. Petitioner presents contentions and evidence concerning why an ordinarily skilled artisan would have been motivated to combine Menon (Ex. 1005), Trefethen (Ex. 1006), RS6000 (Ex. 1007), MPIref (Ex. 1017), and POEref (Ex. 1008) with a reasonable expectation of success to arrive at the limitations of claims 4 and 20. *Id.*

For example, Petitioner contends that “Menon suggests the combination by teaching that MultiMATLAB uses ‘POE, IBM's Parallel Operating Environment,’” and “Menon describes using POE to initialize the MATLAB processes.” Pet. 111 (quoting Ex. 1005, 6). Petitioner also contends that POEref describes that the “Parallel Operating Environment (POE)” provides the command “poe” “to load and execute programs on remote nodes” and “initializ[e] the local environment.” *Id.* at 111–112 (citing Ex. 1008, ix, 28).¹⁷

¹⁷ For findings and analysis of Petitioner’s showing of motivation to combine POEref and Patent Owner’s arguments, *see supra*, § II.D.7.b.

Petitioner further contends as follows with respect to claim 4:

Menon teaches that a user starts a MATLAB process on each processor of the MultiMATLAB architecture by using IBM's POE, an execution environment that includes a program code for loading a program on each one of multiple processor nodes, as taught by POEref, and that each MATLAB process on each non-interactive processor is initialized, includes its MEX routines, a MultiMATLAB interface module, indirection table, and a communication layer (*cluster node module*), rendering obvious this limitation.

Pet. 113 (citing Ex. 1003 ¶¶ 392–397).

Patent Owner does not specifically challenge Petitioner's showing with respect to claim 20. *See* Resp. 19–23. Petitioner's showing for claim 20 is similar to its showing with respect to claim 4, wherein claim 20 is broader in the sense that it does not require a “cluster configuration module.” *See* Pet. 110–114. Regarding “initializes”/“initialization” (claims 4, 20, respectively) Petitioner shows that “POEref describes that the initialization process is started using the ‘poe’ command.” *Id.* at 113 (citing Ex. 1008, 28 (“When you invoke poe, the Partition Manager allocates processor nodes for each task and initializes the local environment”)). In addition, Petitioner shows that “[t]he poe initialization ‘loads your program, and reproduces your local environment, on each processor node.’” *Id.* (quoting Ex. 1008, 28).

Arguing claim 4, Patent Owner contends that “[l]aunching processes on multiple nodes does not initialize cluster node modules as part of a cluster computing architecture; it merely causes programs to begin executing.” Resp. 22. This attorney argument amounts to raising a distinction without a difference. It does not address Petitioner's showing that POEref specifically states that “[w]hen you invoke [POE], the Partition Manager allocates

processor nodes for each task and *initializes* the local environment.” Pet. 113 (quoting Ex. 1008, 28 (emphasis added)); accord Ex. 1163 (explaining how POEref “initializes the local environment” and “loads your program” (quoting Ex. 1008, 28) and “start[s] . . . the MATLAB process and its MultiMATLAB interface/module functionality” (citing Ex. 1005, 3–6).

Patent Owner similarly argues that the POE system “does not configure or initialize application-level modules in the sense required by Claim 4.” Resp. 21. Patent Owner further contends that “any mechanism that merely launches processes or allocates hardware resources” does not satisfy claim 4, because claim 4 “requires an internal module of the claimed system—i.e., a component of the cluster computing architecture itself—that performs initialization of cluster node modules so that they can operate together in the claimed environment.” *Id.* at 19. Patent Owner also argues that POE does not “fill the gap.” *See id.* According to Patent Owner, “POE operates outside the application; it does not configure or initialize application-level modules in the sense required by Claim 4,” because “[t]reating POE as the claimed cluster configuration module collapses external job-launch infrastructure into internal system structure, contrary to the claim language and the patent’s disclosure.” *Id.*

Patent Owner’s arguments imply that claim 4 precludes “[an] external job-launch infrastructure,” which is beyond the scope of claim 4. Claim 4 recites “[t]he computer cluster of claim 1, further comprising a cluster configuration module that initializes at least one of the first cluster node module, the second cluster node module, or the third cluster node module.” Contrary to Patent Owner’s arguments, claim 4 does not specify the “cluster configuration module” is an “internal module of the claimed system” or an “external job-launch infrastructure.” Additionally, as Petitioner argues,

“[t]he ’582 specification . . . explicitly describes that a ‘cluster configuration module’ can be ‘stored . . . on a remote computer system,’ and thus need not be limited to an ‘internal module of the claimed system’ as Patent Owner suggests.” Reply 11–12 (first two quotes quoting Ex. 1001, 11:36–39); *accord* Ex. 1163 ¶ 35 (testifying that such a “remote computer system” module “is not limited to an ‘internal’ module”).

In its Sur-reply, Patent Owner emphasizes that “[t]he operative word in claim 4 is ‘initializes’ — the architectural step that establishes the claimed cluster node modules and their internode software relationships — not the launching, allocating, or program-loading of an operating-system facility.” Sur-reply 7. This argument does not address Petitioner’s showing as outlined above that the POE command specifically describes the function of initializing. *See* Pet. 113 (“Menon teaches that a user starts a MATLAB process on each processor of the MultiMATLAB architecture by using IBM’s POE, an execution environment that includes a program code for loading a program on each one of multiple processor nodes, as taught by POEref, and that each MATLAB process on each non-interactive processor is initialized, includes its MEX routines, a MultiMATLAB interface module, indirection table, and a communication layer (*cluster node module*), rendering obvious this limitation.” (citing Ex. 1003 ¶¶392–397)), 114 (“When you invoke poe, the Partition Manager allocates processor nodes for each task and initializes the local environment” and “[t]he poe initialization ‘loads your program, and reproduces your local environment, on each processor node’” (quoting Ex. 1008, 28)).

Accordingly, Petitioner persuasively shows that claims 4 and 20 would have been obvious.

III. CONCLUSION

We conclude Petitioner has established by a preponderance of the evidence that claims 1–20 of the '582 patent are unpatentable.

In summary:

Claim(s)	35 U.S.C. §	Reference(s)/Basis	Claim(s) Shown Unpatentable	Claim(s) Not Shown Unpatentable
1–3, 5–19	103	Menon, Trefethen, RS6000, MPIref	1–3, 5–19	
4, 20	103	Menon, Trefethen, RS6000, MPIref, POEref	4, 20	
Overall Outcome			1–20	

IV. ORDER

In consideration of the foregoing, it is hereby:

ORDERED that Petitioner has shown by a preponderance of the evidence that claims 1–20 of the '582 patent are unpatentable;

FURTHER ORDERED that, because this is a Final Written Decision, parties to the proceeding seeking judicial review of the decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.¹⁸

¹⁸ Should Patent Owner wish to pursue amendment of the challenged claims in a reissue or reexamination proceeding subsequent to the issuance of this decision, we draw Patent Owner's attention to the April 2019 *Notice Regarding Options for Amendments by Patent Owner Through Reissue or Reexamination During a Pending AIA Trial Proceeding*. See 84 Fed. Reg. 16,654 (Apr. 22, 2019). If Patent Owner chooses to file a reissue application or a request for reexamination of the challenged patent, we remind Patent

IPR2025-00916
Patent 11,811,582 B2

FOR PETITIONER:

Andrew Ehmke
David McCombs
Dan Smith
Dagim Tilahun
HAYNES AND BOONE, LLP
andy.ehmke.ipr@haynesboone.com
david.mccombs.ipr@haynesboone.com
dan.smith.ipr@haynesboone.com
dagim.tilahun.ipr@haynesboone.com

Brian Ferguson
Chaoxuan Liu
WINSTON & STRAWN LLP
beferguson@winston.com
ccliu@winston.com

James Kappos
jwk92657@gmail.com

FOR PATENT OWNER:

David Lindner
Jon Gurka
CROWELL & MORING, LLP
dlindner@crowell.com
jgurka@crowell.com

Reynaldo Barcelo
BARCELO, HARRISON & WALKER, LLP
rey@bhiplaw.com

Owner of its continuing obligation to notify the Board of any such related matters in updated mandatory notices. *See* 37 C.F.R. § 42.8(a)(3), (b)(2).