

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

INTEL CORPORATION,

Petitioner,

v.

ADVANCED CLUSTER SYSTEMS, INC.,

Patent Owner.

IPR2025-00913

U.S. Patent No. 11,570,034

PATENT OWNER RESPONSE

TABLE OF CONTENTS

I.	Introduction.....	1
A.	The '034 Patent And The Claimed Distributed Computing Architecture	1
B.	The Narrow, Dispositive Issues That Control This Case.....	3
C.	Petitioner Cannot Meet Its Burden On Any Instituted Ground	6
II.	Procedural Background and Governing Legal Standards	7
A.	Instituted Grounds and Challenged Claims.....	7
B.	Petitioner's Burden of Proof Never Shifts	7
C.	Limits on Expert Testimony and Impermissible Gap-Filling	8
D.	Reliance On The Existing Evidentiary Record	8
III.	Claim Construction.....	9
A.	Governing Claim Construction Principles	9
B.	“Hardware Processor with a Plurality of Processing Cores” (Limitations [1.4.1], [1.5.1]) / “Hardware Processor Comprises Multiple Processor Cores” (Limitation [30.1.2])	10
1.	The Claim Language Requires a Single Multi-Core Processor ...	11
2.	The Specification Confirms The Claimed Multi-Core, Not Multi-Processor, Architecture.....	12
3.	Expert and Technical Evidence Confirms the Claimed Architecture Requires a Single Multi-Core Processor—Not SMP.....	14
IV.	Petitioner Fails to Establish a <i>Prima Facie</i> Case of Obviousness	16
A.	The Asserted Prior Art Does Not Disclose A Multi-Core Processor 17	
B.	Petitioner's “Eval-Return” Theory Does Not Satisfy The Claim Requirements.....	21
1.	The Claims Require Programmatic Return of Evaluation Results.....	24

2.	Trefethen's Eval () Command Does Not Return Results But Rather Forwards Screen Output Only	26
3.	Data Retrieval Requires Separate Get/Send Operations	27
4.	Menon Adds No Return-of-Results Mechanism.....	28
5.	Dr. Bajaj's Eval-Return Assertions Are Conclusory and Unsupported.....	31
C.	The Asserted Prior Art Does Not Teach The Claimed Three-Node Execution Chain	32
1.	The Claims Require a Specific 1→2→3→1 Order of Operations	34
2.	No Reference Discloses or Suggests This Architecture.....	35
3.	Petitioner's "Could Have" Scenarios Are Legally Insufficient ...	38
4.	Consistency with Prior PTAB Decisions Rejecting Similar Theories.....	40
V.	No Motivation to Combine and No Reasonable Expectation of Success	41
A.	Petitioner Offers Only a General Goal, Not a Record-Based Motivation to Build the Claimed System.....	41
B.	Menon Is A Redesign, Not An Extension, Of Trefethen	42
C.	Menon Identifies Fundamental Deficiencies In The Earlier Architecture	43
D.	The Proposed Combination Would Not Yield The Claimed System	45
E.	Petitioner's Theory Relies on Impermissible Hindsight.....	47
VI.	Menon Teaches Away from Petitioner's Proposed Combinations	48
A.	Express Criticism of the First-Generation MultiMATLAB Architecture	49
B.	Incompatibility With Parallel MEX Extensions	51
C.	Teaching Away Undermines Any Motivation to Combine	52
VII.	Petitioner's Expert Testimony Cannot Cure Missing Claim Elements.....	53
A.	Conclusory Assertions Are Not Substantial Evidence.....	54

B.	Conflicts with the Written Record Must Be Resolved Against Petitioner	55
C.	Petitioner Is Bound by the Theories in the Petition	57
VIII.	Petitioner’s References Are Not “Printed Publications”	58
A.	IBM Technical Manuals (RS/6000 and POEref, Exs. 1007–1008)	59
B.	MPI Forum Draft Standard (“MPIref,” Ex. 1017)	60
C.	Cornell “MultiMATLAB” Web Article (“Trefethen,” Ex. 1006)	61
D.	SC’97 Conference Paper (“Menon,” Ex. 1005)	63
E.	Petitioner’s Citation-Based Arguments Will Be Addressed Separately	65
IX.	Preserved Evidentiary Issues	66
X.	Conclusion	66

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>ActiveVideo Networks, Inc. v. Verizon Commc'ns, Inc.</i> , 694 F.3d 1312 (Fed. Cir. 2012)	38
<i>Arendi S.A.R.L. v. Apple Inc.</i> , 832 F.3d 1355 (Fed. Cir. 2016)	<i>passim</i>
<i>Argentum Pharm. LLC v. Research Corp. Techs.</i> , IPR2016-00204, Paper 19 (PTAB May 23, 2016)	61
<i>Baran v. Med. Device Techs., Inc.</i> , 616 F.3d 1309 (Fed. Cir. 2010)	14
<i>Blue Calypso, LLC v. Groupon, Inc.</i> , 815 F.3d 1331 (Fed. Cir. 2016)	58, 60, 63
<i>In re Cronyn</i> , 890 F.2d 1158 (Fed. Cir. 1989)	60
<i>In re Cyclobenzaprine Hydrochloride Extended-Release Capsule Patent Litig.</i> , 676 F.3d 1063 (Fed. Cir. 2012)	46
<i>In re Dembiczak</i> , 175 F.3d 994 (Fed. Cir. 1999)	47
<i>DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.</i> , 567 F.3d 1314 (Fed. Cir. 2009)	44, 48, 50, 51
<i>In re Gurley</i> , 27 F.3d 551 (Fed. Cir. 1994)	48, 50, 52
<i>Intercontinental Great Brands LLC v. Kellogg N. Am. Co.</i> , 869 F.3d 1336 (Fed. Cir. 2017)	47
<i>In re Klopfenstein</i> , 380 F.3d 1345 (Fed. Cir. 2004)	61

IPR2025-00913 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,570,034

<i>KSR Int’l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007).....	41
<i>Kyocera Wireless Corp. v. Int’l Trade Comm’n</i> , 545 F.3d 1340 (Fed. Cir. 2008)	58
<i>In re Lee</i> , 277 F.3d 1338 (Fed. Cir. 2002)	38, 39, 55
<i>In re Lister</i> , 583 F.3d 1307 (Fed. Cir. 2009)	58
<i>In re Magnum Oil Tools Int’l, Ltd.</i> , 829 F.3d 1364 (Fed. Cir. 2016)	<i>passim</i>
<i>Medtronic, Inc. v. Barry</i> , 891 F.3d 1368 (Fed. Cir. 2018)	58, 60, 65
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005) (en banc)	9
<i>Procter & Gamble Co. v. Teva Pharms. USA, Inc.</i> , 566 F.3d 989 (Fed. Cir. 2009)	41
<i>Samsung Elecs. Co. v. Infobridge Pte. Ltd.</i> , 929 F.3d 1363 (Fed. Cir. 2019)	59, 61, 62
<i>TIP Sys., LLC v. Phillips & Brooks/Gladwin, Inc.</i> , 529 F.3d 1364 (Fed. Cir. 2008)	14
<i>TQ Delta, LLC v. Cisco Sys., Inc.</i> , 942 F.3d 1352 (Fed. Cir. 2019)	8, 24, 32, 54
<i>Upjohn Co. v. Mova Pharm. Corp.</i> , 225 F.3d 1306 (Fed. Cir. 2000)	21
Statutes	
35 U.S.C. § 311(b)	5
35 U.S.C. § 316(e)	7, 53

IPR2025-00913 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,570,034

37 C.F.R. § 42.6(e).....	69
37 C.F.R. § 42.23(b)	57
37 C.F.R. § 42.24	68
37 C.F.R. § 42.24(d)	68
37 C.F.R. § 42.65(a).....	21, 32, 54

PATENT OWNER’S EXHIBIT LIST

<u>Exhibit</u>	<u>Description</u>
2001	DocketNavigator Statistics for Motion Success for Stay Pending IPR (Post-Institution) for Judge Albright
2002	J. Albright Standing Order Governing Proceedings (OGP) 4.4 – Patent Cases (Jan 23, 2024)
2003	DocketNavigator Statistics for Time-to-Milestones for Judge Albright
2004	Advanced Cluster Systems, Inc. v. Intel Corporation, Amended Scheduling Order (D.I. 54)
2005	Advanced Cluster Systems, Inc. v. Advanced Micro Devices, Inc., Scheduling Order (D.I. 36)
2006	Judge Albright Clerk emails
2007	Allani v. Apple Inc., No. 6:24-cv-304-ADA (W.D. Tex. May 6, 2025) (D.I. 30)
2008	Intellectual Ventures I LLC v. TCL Elects. Holdings Ltd., No. 6:23-cv-309-ADA (W.D. Tex. Sept. 10, 2024) (D.I. 44)
2009	Acquis, LLC v. Hon Hai Precision Indus Co. Ltd., No. 6:23-cv-264-ADA (W.D. Tex. May 31, 2024) (D.I. 46)
2010	ParkerVision, Inc. v. MediaTek Inc. et al., No. 6:22-cv-01163-ADA (W.D. Tex. April 29, 2024)
2011	TOT Power Control, S.L. v. AT&T, Inc. et al., No. 6:21-cv-00107-ADA (W.D. Tex. June 6, 2025)

IPR2025-00913 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,570,034

2012	AlmondNet, Inc. v. Amazon.com, Inc., et al., No. 6:21-cv-00898-ADA (W.D. Tex. June 19, 2025)
2013	Intel Corporation’s Amended Invalidity Contention Cover Pleading (Apr. 25, 2025)
2014	[EXHIBIT NOT USED]
2015	AMD’s Preliminary Invalidity & Subject-Matter Eligibility Contentions Cover Pleading (Corrected Apr. 25, 2025)
2016	AMD’s Preliminary Invalidity Contention – Chart A-8
2017	Declaration of Dr. Melissa C. Smith Under 37 C.F.R. § 1.68
2018	Curriculum Vitae of Dr. Melissa C. Smith
2019	Advanced Cluster Systems, Inc. v. Intel Corporation, Order Appointing Technical Advisor (D.I. 71)
2020	Advanced Cluster Systems, Inc. v. Advanced Micro Devices, Inc., Order Appointing Technical Advisor (D.I. 53)
2021	FAQ: The Sustainable Data Center – More Energy Efficient AI (available at https://www.intel.com/content/www/us/en/contentdetails/791405/faq-the-sustainable-data-center-ai-usecase.html?wapkw=sustainability&DocID=791405)
2022	NVIDIA Company Overview (May 2025) (available at https://investor.nvidia.com/events-andpresentations/presentations/presentation-details/2025/Company-Overview-5-25/default.aspx)

IPR2025-00913 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,570,034

- 2023 Intel Corporation's Answer and Counterclaims to First Amended Complaint, Advanced Cluster Systems, Inc. v. Intel Corporation, 7:24-cv-00245-ADA (W.D. Tex. Feb. 5, 2025) (D.I. 41)
- 2024 Exhibit A-18 - MultiMatlab References, U.S. Patent No. 10,333,768, as listed in Defendant NVIDIA Corporation's Supplemental Invalidity Contentions in Case No. 1 :19-cv-2032-CFC, Advanced Cluster Systems, Inc., v. NVIDIA Corporation as dated August 17, 2021 in 254 pages (submitted with IDS in U.S. Patent Appln. No. 17/374,789) (ACS_0065212-0065465)
- 2025 Advanced Cluster Systems, Inc. v. Intel Corporation, Case No. 7:24-cv-00245-ADA (D.I. 79), Minute Entry Markman Hearing Proceedings Held 08/18/2025
- 2026 VirtaMove, Corp. v. Microsoft Corp., Case No. 1:25-cv-00794-ADA (D.I. 33), Order Granting Motion to Dismiss
- 2027 Yeti Coolers, LLC v. Waterbear Global, LLC, Case No. 1:24-cv-1251-ADA (D.I. 35, 35-1), Unopposed Motion to Extend Scheduling Order Deadlines and Proposed Order
- 2028 Intel Corporation's Third Amended Preliminary Invalidity And Subject-Matter Eligibility Contentions (Aug. 28, 2025)
- 2029 (New) Patent Owner's Objections Under 37 C.F.R. § 42.64(b)(1) (Served Nov. 21, 2025)

I. INTRODUCTION

Patent Owner Advanced Cluster Systems, Inc. (“ACS”) hereby files this Response to the Petition in IPR2025-00913 challenging claims 1–30 of U.S. Patent No. 11,570,034 (“the ’034 Patent”). Petitioner contends that claims 1–25, 27–28, and 30 would have been obvious under pre-AIA § 103 over Menon in view of Trefethen, RS6000, and POEref, and that claims 26 and 29 would have been obvious further in view of MPIref. The Petition’s obviousness combinations cannot withstand merits scrutiny. Institution was based on a preliminary record and an express order for further claim construction, with a final determination made on the full record developed at trial. On that full record, Petitioner cannot carry its burden to prove unpatentability by a preponderance of the evidence.

A. The ’034 Patent And The Claimed Distributed Computing Architecture

The ’034 Patent is directed to cluster computing systems that provide cluster computing functionality to high-level computing environments—particularly environments in which users express computations as mathematical expressions and that depend on interactive workflows. The patent distinguishes cluster computing from multi-node distributed systems that merely run separate jobs on separate nodes without peer-level intercommunication between nodes during computation. In the claimed systems, nodes perform coordinated operations as specified by the claim

language, including communication steps required to carry out the recited multi-stage evaluation. (*See, e.g.*, Ex. 1001, 29:41-43.)

The patent's disclosed architecture includes, *inter alia*, a user interface module, cluster node modules, and single-node kernel modules. (*See, e.g.*, Ex. 1001, Fig. 2.) The kernel modules execute single-threaded program code (tasks) and evaluate mathematical expressions (*id.* 23:1-18); the cluster node modules provide peer-to-peer communications among kernel modules while tasks execute (*id.* 11:44-22:17, 23:51-52, 24:39-40, 24:52-53, 25:1-2, 25:9-10, 25:23-24, 25:37-38); and the user interface module enables users to issue instructions into the system and receive results (*id.* 22:41-67). The patent's figures and written description describe a peer-to-peer communication framework between nodes, in which results of mathematical expression evaluations are communicated back through the cluster node modules to the user interface module. (Ex. 1001, Fig. 2, 4, 5; *id.* 11:44-22:17, 23:1-18, 23:51-52, 24:39-40, 24:52-53, 25:1-2, 25:9-10, 25:23-24, 25:37-38, 22:41-67.) The claims recite that architecture by requiring (among other things) (i) peer-to-peer communication of evaluation results, (ii) a specific chained evaluation flow across multiple nodes, and (iii) multi-core hardware processors at designated nodes. (*See, e.g.*, Ex. 1001, 29:41-30:3; *id.* 31:55-32:19; *id.* 32:56-33:26; *id.* 34:19-50.)

B. The Narrow, Dispositive Issues That Control This Case

This case turns on a small number of dispositive issues that run through every instituted ground and every challenged independent claim.

First, Petitioner cannot establish the prior art teaches the multi-core processor limitations in every claim when properly construed. Independent Claims 1, 25, and 28 require that the second and third nodes comprise “a hardware processor with a plurality of processing cores,” and independent claim 30 also requires that “the hardware processor comprises multiple processor cores.”¹ (Ex. 1001, 29:52-53; 29:59-60; *id.* 32:4-5; 32:10-11; *id.* 33:9-10; 33:15-16; *id.* 34:14-15; 34:39-41.) Petitioner’s only mapping for these limitations relies on RS6000’s description of a SMP node populated with multiple discrete processors by attempting to relabel those processors as “cores.” That relabeling is inconsistent with the claim language, the processor/core distinction in the ’034 Patent specification, and RS6000’s own description of a SMP “configuration,” and it is refuted by Dr. Smith’s testimony (Ex.

¹ During claim construction in the parallel litigation, the Court found that the phrase “communicate a result of the second mathematical expression evaluation to the first node” in claim 30 of the ’034 Patent to be indefinite. Ex. 1085 at 16. Indefiniteness is not a ground before the Board, and because the Court’s claim construction is still appealable, ACS explains herein why Petitioner’s relied upon prior art does not disclose “the hardware processor comprises multiple processor cores” required by claim 30.

2017). Because this multi-core limitation is required by every challenged claim, the lack of disclosure in Petitioner's relied-upon prior art is dispositive in this case.

Second, Petitioner's case depends on an "Eval-return" premise that the asserted references do not disclose to satisfy the [1.6], [25.7], [28.8] and [30.1.1] limitations of the independent claims. Petitioner attempts to equate forwarded screen output and user-visible display with the claims' requirement that evaluation results are returned to the first node and then to the user interface as part of the system's programmed operation. However, neither Trefethen nor Menon support that equivalence. Petitioner's mapping therefore rests on expert narrative rather than disclosure in the references.

Third, Petitioner does not establish the claimed three-node execution flow. Claims 1, 25, 28, and 30 require a specific $1 \rightarrow 2 \rightarrow 3 \rightarrow 1$ order of operations: the second node evaluates a first expression and communicates its result to the third node; the third node evaluates a second expression using that received result and communicates the result back to the first node; and the first node returns that result to the user interface. Petitioner relies on a user-scripted demonstration and an undisclosed "could have" modification of the references to approximate this claimed architecture, but no reference discloses nodes configured to perform the required three-node execution flow as claimed.

Fourth, even if Petitioner could overcome these disclosure failures, it has not established a legally sufficient motivation to combine the asserted references or a reasonable expectation of success. As explained in Sections V and VI, Menon itself identifies structural limitations in the earlier MultiMATLAB architecture and presents a redesign rather than an extension, teaching away from the modifications Petitioner proposes. Petitioner's motivation theory relies on generalized goals and hindsight reconstruction, not record-based reasons tied to the claimed architecture.

Fifth, Petitioner's case depends on expert narrative that cannot substitute for what the prior art actually discloses. As explained in Section VII, Petitioner relies on conclusory expert assertions to bridge gaps in the asserted references, including gaps concerning result return, execution flow, and architectural configuration. Where the written record is silent or inconsistent with Petitioner's theory, expert ipse dixit cannot supply the missing claim limitations or rewrite the disclosed system to match the claims.

Sixth, Petitioner has failed to establish that certain asserted references qualify as printed publications as a merits issue under 35 U.S.C. § 311(b). As discussed in Section VIII, Petitioner has not shown that those references were publicly accessible at the relevant time, instead relying on inference, secondary citations, or post hoc evidence rather than contemporaneous proof of dissemination. That failure

independently defeats Petitioner's reliance on those references for its obviousness grounds.

Finally, Patent Owner has preserved evidentiary objections that further undermine the weight of Petitioner's proof. As addressed in Section IX, Patent Owner timely objected to deficiencies in authentication, foundation, and Petitioner's reliance on unsupported secondary materials. Even if the Board were to reach these issues separately, they provide additional grounds for declining to credit Petitioner's evidence or expert testimony. Any one of these proof defects defeats Petitioner's instituted grounds. Together, they demonstrate that Petitioner cannot carry its burden on any challenged claim.

C. Petitioner Cannot Meet Its Burden On Any Instituted Ground

Petitioner bears the burden to prove unpatentability by a preponderance of the evidence, and the burden never shifts. Petitioner must therefore establish that the asserted references disclose each required limitation under the proper claim constructions and that any asserted combination is supported by a legally sufficient rationale. As shown below, Petitioner's instituted grounds fail because they depend on: (1) relabeling RS6000's SMP disclosure to satisfy the processor multi-core limitations; (2) conflating screen output with programmatic return of evaluation results; and (3) supplying a missing three-node execution flow through inference and

expert *ipse dixit* rather than disclosure. The sections that follow address these failures directly in the order that resolves the proceeding most efficiently.

II. PROCEDURAL BACKGROUND AND GOVERNING LEGAL STANDARDS

A. Instituted Grounds and Challenged Claims

Intel Corporation (“Petitioner”) is now the only petitioner in this proceeding. (Paper 21, 2.) The Board instituted *inter partes* review of claims 1–30 of the ’034 Patent and instituted on all asserted grounds. (Paper 16, 49.) The instituted grounds are: (i) claims 1–25, 27–28, and 30 as allegedly obvious over Menon in view of Trefethen, RS6000, and POEref; and (ii) claims 26 and 29 as allegedly obvious over Menon in view of Trefethen, RS6000, POEref, and in further view of MPIref. (*Id.* at 10, 49.) The Board ordered the parties to propose and support a particularized construction of limitations [1.4.1], [1.5.1], and [30.1.2] and to analyze the relevant claim language and cite the specification and prosecution history, with particular attention to the patent’s description of processors and processor cores. (*Id.*, 14.) Patent Owner addresses those claim construction issues in Section III.

B. Petitioner’s Burden of Proof Never Shifts

Petitioner bears the burden of proving unpatentability by a preponderance of the evidence, and that burden never shifts to Patent Owner. 35 U.S.C. § 316(e); *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1375 (Fed. Cir. 2016). Accordingly,

Petitioner must establish that the asserted references disclose each required limitation under the proper constructions and that any asserted combination is supported by a legally sufficient rationale.

C. Limits on Expert Testimony and Impermissible Gap-Filling

The Board's merits determinations must rest on evidence, not conclusory expert narrative used to supply missing disclosures. Conclusory expert testimony is not substantial evidence. *TQ Delta, LLC v. Cisco Sys., Inc.*, 942 F.3d 1352, 1359–60 (Fed. Cir. 2019). And neither “common sense” nor expert intuition may be used to fill a missing claim element where the prior art is silent on the specific limitation at issue. *Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1362–63 (Fed. Cir. 2016). These principles apply with particular force where, as here, Petitioner's theories depend on re-labeling a prior art SMP configuration to satisfy the multi-core processor limitations, and on attributing result-return and chained evaluation behaviors to prior art references that do not disclose these claimed requirements.

D. Reliance On The Existing Evidentiary Record

Patent Owner does not submit a new expert declaration with this Response because it is not necessary for the Board to resolve the issues presented. The dispositive issues are resolved by the intrinsic record and the asserted references. To the extent expert testimony is helpful on the multi-core processor limitation that runs

through the entire challenged claim set, Patent Owner relies on Dr. Smith's existing declaration (Ex. 2017), which addresses Petitioner's RS6000 SMP mapping and the absence of any disclosure of a hardware processor comprising multiple cores. The remainder of Patent Owner's rebuttal relies on direct quotations from the references and record citations showing that Petitioner's theories depend on mischaracterizations of the references and on impermissible gap-filling.

III. CLAIM CONSTRUCTION

A. Governing Claim Construction Principles

The Board applies the *Phillips* standard. Claim terms are given their ordinary and customary meaning as understood by a POSITA at the time of the invention, in light of the intrinsic record. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–17 (Fed. Cir. 2005) (en banc). Here, the Board has ordered the parties to provide a particularized construction of limitations [1.4.1], [1.5.1], and [30.1.2], with particular attention to the '034 Patent's description of processors and processor cores. The dispute therefore turns on the claim language and the patent's own processor/core distinction in the specification—not on generalized assertions about what multiprocessor systems can do.

B. “Hardware Processor with a Plurality of Processing Cores” (Limitations [1.4.1], [1.5.1]) / “Hardware Processor Comprises Multiple Processor Cores” (Limitation [30.1.2])

The Board has identified a dispute over the plain meaning of these limitations, and has ordered the parties to propose and support a particularized construction grounded in the claim language and the specification, with particular attention to the patent’s description of processors and processor cores. (Paper 16 at 13–15; citing Ex. 1001, 5:11–37, 7:14–36, 9:50–10:3.)

Petitioner’s instituted grounds depend on an RS6000-based mapping that equates an IBM SP2 “Symmetric MultiProcessor (SMP)” configuration containing multiple discrete processors with the claim’s express requirement of a “hardware processor with a plurality of processing cores.” (Pet. 43; Ex. 1003 ¶¶ 207–210, 216–217.)

Petitioner’s construction is unsupported by the claim language, inconsistent with the ’034 Patent’s specification, and contradicted by RS6000 itself. Under the proper construction, RS6000 does not disclose the required multi-core processor limitation, and that failure is dispositive because every challenged claim recites the same requirement. (Ex. 2017 ¶ 25.)

1. The Claim Language Requires a Single Multi-Core Processor

The claim language itself compels a processor-centric, multi-core reading. Limitations [1.4.1], [1.5.1], as well as [25.5.1], [25.6.1], [28.6.1] and [28.7.1], do not recite a “node with multiple processors.” They recite a “node comprising a hardware processor with a plurality of processing cores.” (Ex. 1001, 29:52–53, 59–60; *id.* 32:4-5, 32:10-11; *id.* 33:9-11, 15-16.) The grammar matters: the claims require a singular “hardware processor” that “comprises” (or “with”) multiple “processing cores.”

Claim 30 [30.1.2] confirms the same meaning in slightly different words. It recites “wherein the hardware processor comprises multiple processor cores” and also recites the same “hardware processor with a plurality of processing cores” language. (Ex. 1001, 34:14–15, 34:40–41.)

The internal structure of claim 1 reinforces the drafters’ deliberate choice to specify “plurality of processing cores” only where they intended it. Claim 1 describes a first node “comprising a first hardware processor” configured to access memory containing code for a user interface and a first single-node kernel. (Ex. 1001, 29:45–51.) It then recites a second node and third node, each expressly requiring a “hardware processor with a plurality of processing cores.” (*Id.*, 29:52–53, 59–60.)

Claim 1 thus draws an intentional distinction among the nodes' processor requirements. Petitioner's construction would obliterate that distinction by treating any node-level SMP arrangement as satisfying "hardware processor with a plurality of processing cores," which collapses "processor" and "node" and renders the "cores" language functionally meaningless. Claim construction does not permit reading words out of the claim to fit the prior art.

2. The Specification Confirms The Claimed Multi-Core, Not Multi-Processor, Architecture

The Board has directed the parties to the specification's discussion of processors and processor cores. That specification confirms that the patent uses "processor" and "cores" in a conventional multi-core processor configuration, not as a synonym for a board-level SMP configuration with multiple processors. The specification explains: "Each processor can contain one or more processor cores that are capable of single-threaded execution. Processor cores are generally independent microprocessors, but more than one can be included in a single-chip package." (Ex. 1001, 5:11–16.)

This disclosure does two critical things at once. First, it distinguishes the processor from the cores it contains. Second, it expressly locates multiple cores within a "single-chip package."

The specification further reinforces this understanding by expressly contemplating “single-core or multi-core microprocessors.” (Ex. 1001, 7:15–25.) The patent does not say “single-processor or multi-processor nodes” in these passages. And when the patent gives a concrete example of a multi-core processor, it cites a Core 2 Duo® processor as having “two processing cores.” (*Id.* 7:32–33.)

That example is dispositive of how the patent uses “processor” and “cores”: a Core 2 Duo is one processor package with two cores. *See infra* at III.B.3 below. It is not two processors on a board. The patent’s specification thus anchors “plurality of processing cores” to the multi-core processor architecture that Petitioner’s references fail to disclose.

Petitioner has attempted to turn the specification’s statement that “processor cores are generally independent microprocessors” against Patent Owner by arguing that this encompasses an SMP configuration of multiple single-core processors on separate chips. (Paper 9, 30; Paper 16 at 13–15.) But that argument misunderstands the specification and ignores the very next clause: “but more than one can be included in a single-chip package.” (Ex. 1001, 5:14–16.)

Therefore, the specification explains that a “core” can be microprocessor-like in capability, but when multiple such processing units are integrated within a single chip/package, they are treated as “cores” of that processor. That is the multi-core

concept. Petitioner’s proposed reading strips the “single-chip package” language of meaning and collapses the processor/core distinction the patent carefully draws.

Moreover, even if some disclosure in the specification could be read to describe broader hardware implementations, it does not follow that these particular claim limitations must be construed to capture that breadth. It is well established that when a construction is supported by the intrinsic evidence that “the claims of the patent need not encompass all disclosed embodiments” if including other embodiments would contradict the plain language of the claim, and it is common for different claims to cover different embodiments. *TIP Sys., LLC v. Phillips & Brooks/Gladwin, Inc.*, 529 F.3d 1364, 1373 (Fed. Cir. 2008); *Baran v. Med. Device Techs., Inc.*, 616 F.3d 1309, 1316 (Fed. Cir. 2010). Here, the claim language explicitly reads as multi-core within a processor, the specification confirms that usage, and Petitioner’s alternative construction would rewrite the claims to fit RS6000 rather than apply the claims against RS6000.

3. Expert and Technical Evidence Confirms the Claimed Architecture Requires a Single Multi-Core Processor—Not SMP

Petitioner’s effort to equate RS/6000’s SMP configuration with a “hardware processor with a plurality of processing cores” is not only incompatible with the

claim language and the specification—it is also directly refuted by the technical evidence and expert testimony in the record.

As Dr. Smith explains, a person of ordinary skill in the art would understand that the claim language—“a hardware processor with a plurality of processing cores”—refers to a single processor package containing multiple cores. (Ex. 2017 ¶¶ 26, 33.) That is the ordinary usage of the term “multi-core processor” at the time of the invention. (*Id.*) It is not how symmetric multi-processing (SMP) is described. SMP refers to a system-level configuration in which a node contains multiple physically separate processors, each typically single-core, operating in parallel. (*Id.* ¶ 27; *see also* Ex. 1007, 2, 9.) If the patent had intended to encompass that arrangement, it would have used different language—such as “plurality of processors” or “multiple processors.” It did not. (Ex. 1001, 29:52–53, 59–60; 34:14–15, 34:40–41.)

Dr. Smith further confirms that the challenged claims are consistent in their terminology. Claims 1, 25, 28, and 30 all require a “hardware processor with a plurality of processing cores,” and the dependent claims incorporate that limitation through dependency. (Ex. 2017 ¶ 25.) The record confirms that a POSITA would not treat this language as encompassing SMP hardware.

Dr. Smith confirms that a Core 2 Duo is a single processor chip with two integrated cores—not two processors on a board. (Ex. 2017 ¶¶ 26, 33.) That disclosure and its expert interpretation anchor the claim language to a specific hardware model—one that Petitioner’s SMP-based mapping does not fit.

The expert evidence confirms how a POSITA would interpret that language. Petitioner’s attempt to shoehorn SMP nodes into that limitation is legally and technically incorrect.

IV. PETITIONER FAILS TO ESTABLISH A *PRIMA FACIE* CASE OF OBVIOUSNESS

Petitioner advances two instituted obviousness grounds. Ground 1 alleges that claims 1–25, 27–28, and 30 would have been obvious over Menon in view of Trefethen, RS6000, and POEref. Ground 2 alleges that claims 26 and 29 would have been obvious over the same combination in further view of MPIref. Although the grounds add references and vary by claim group, they share the same core theory: that MultiMATLAB (Menon/Trefethen), implemented on IBM SP hardware (RS6000) and launched through IBM’s parallel environment (POEref), discloses or renders obvious the ’034 Patent’s claimed peer-to-peer cluster computing architecture for evaluating mathematical expressions and returning results to a user interface.

Those instituted theories fail for multiple independent reasons. First, the asserted combination does not disclose the Board-ordered multicore processor limitations—“hardware processor with a plurality of processing cores” and “hardware processor comprises multiple processor cores”—because RS6000 describes an SMP node populated with multiple discrete single-core processors, not a multi-core processor as required by the claims, confirmed by the intrinsic record, and as understood by a POSITA (Ex. 2017 ¶¶ 24-36). Second, Petitioner’s mapping depends on conflating screen output with programmatic return of evaluation results to the originating node and user interface. Third, the asserted art does not disclose the claimed chained three-node order of operations (1→2→3→1), as Petitioner’s references fail to disclose nodes configured to execute the required dependent evaluation chain and return path as claimed.

The sections that follow address these structural defects in the order that resolves the instituted grounds most directly.

A. The Asserted Prior Art Does Not Disclose A Multi-Core Processor

As shown in Section III, the properly construed claims require a single hardware processor that includes multiple processor cores, consistent with the patent’s processor/core distinction and “single-chip package” discussion. Under that construction, RS6000’s SMP “configuration”—a node populated with multiple

discrete single-core processors—does not disclose the required multicore processor limitation, as confirmed by Dr. Smith (Ex. 2017 ¶¶ 24-36). Because every challenged claim requires this limitation, Petitioner’s instituted grounds fail for lack of proof on this element alone. (Ex. 2017 ¶¶ 25, 28–36.)

Petitioner’s mapping is explicit: it contends RS6000 teaches that each node has an “SMP” (which Petitioner labels the “hardware processor”) and that each SMP includes “four, eight, twelve, or sixteen ... processors” (which Petitioner labels “plurality of processing cores”), and therefore “each node has a hardware processor with a plurality of processing cores.” (Pet. 43; Paper 16 at 30–32.)

That mapping fails at both the linguistic level and the technical level.

At the linguistic level, Petitioner improperly treats an “SMP configuration” as if it were a “hardware processor.” RS6000 states that nodes have either “a Symmetric MultiProcessor (SMP) configuration or a uniprocessor configuration.” (Ex. 1007, 2; quoted and discussed in Ex. 2017 ¶ 29.)

As Dr. Smith explains, “configuration” signifies an architectural layout of multiple hardware processors working together, not a description of a single hardware processor comprising multiple cores. (Ex. 2017 ¶¶ 28–29.)

Petitioner’s mapping simply erases the word “configuration” and then treats the resulting phrase as a unitary “hardware processor.” By doing so, Petitioner

ignores the actual claim language with a selective quotation from the prior art to rewrite the claim.

At the technical level, RS6000 does not disclose multi-core processors. It discloses multiple discrete processors per node. RS6000 states that each node “ha[s] four, eight, twelve, or sixteen ... processors per node.” (Ex. 1007, 9; discussed in Ex. 2017 ¶ 30.)

It further explains that the relevant card includes “[f]our processor slots allowing a maximum of sixteen processors per node,” that a minimum of four processors are “mounted on one card,” and that additional four-processor cards can be added to reach “a total of sixteen CPUs.” (Ex. 1007, 9–10; Ex. 2017 ¶ 31.)

That configuration is an SMP node: the node contains multiple separate, single-core CPU chips installed on one or more processor cards, rather than a single processor chip that includes multiple cores. It says nothing about the number of processing cores per hardware processor. (Ex. 2017 ¶¶ 30–31.)

Dr. Smith addresses this directly. She explains that RS6000 “never states that any individual hardware processor has multiple processing cores” and that it “describes SMP as a configuration of multiple discrete hardware processors.” (Ex. 2017 ¶ 28.)

She further concludes that Dr. Bajaj’s assertion that RS6000 discloses “a hardware processor with a plurality of processing cores” is “unfounded,” based on “a misreading and partial quotation of the RS/6000 reference,” and is contrary to the context showing that POWER3 is a single-core processor. (Ex. 2017 ¶ 32.)

Petitioner’s attempt to turn multiple discrete processors into “processing cores” therefore fails on the evidence.

Legally, Petitioner’s approach is also untenable because it rewrites claim language to accommodate the prior art rather than applying the claim language to the prior art. The patent uses “processing cores” as part of a multi-core processor and distinguishes cores from other microprocessors by emphasizing that multiple cores can be included in a “single-chip package.” (Ex. 1001, 5:14–16.)

Finally, Petitioner’s theory suffers from a dispositive proof failure: RS6000’s discussion of “processors per node” cannot establish the required “cores per processor” limitation. Petitioner has offered no documentary evidence that the POWER3 processors in RS6000 were multi-core, and Dr. Smith affirmatively establishes the opposite—that POWER3 was single-core. (Ex. 2017 ¶ 32.)

Petitioner’s “SMP equals multicore processors” position therefore rests entirely on a conclusory expert declaration, which cannot create prior art disclosure

that is not there. (37 C.F.R. § 42.65(a); *Upjohn Co. v. Mova Pharm. Corp.*, 225 F.3d 1306, 1311 (Fed. Cir. 2000).)

The proper construction of limitations [1.4.1], [1.5.1], and [30.1.2] is that they require a single multi-core processor. Under that proper construction, Petitioner cannot satisfy the limitation with RS6000's SMP node configuration, and because every challenged claim recites the same requirement, Petitioner's instituted obviousness grounds fail.

B. Petitioner's "Eval-Return" Theory Does Not Satisfy The Claim Requirements

In this section, "Eval-return" refers to Petitioner's theory—not the claim language—that MultiMATLAB's `Eval()` command, along with its screen display behavior, satisfies the return-of-results functionality required by claims 1, 2, 25, 28, and 30. Although Petitioner does not label it this way, the term is used as shorthand for Petitioner's repeated reliance on Eval-based output as a substitute for a system-level return of computed results.

Petitioner invokes this theory in attempting to satisfy the following limitations:

- Claim 1 requires that the third node "communicate the result of the second mathematical expression evaluation to the first node," and that the first node is

“configured to return the result of the second mathematical expression evaluation to the user interface.” (Ex. 1001, 29:65-30:3.)

– Claim 2, which depends on claim 1, further requires that the result be returned “without transmitting the result through a master node.” (*Id.* 30:8–10.)

– Claim 25, which requires that “the third node is configured to return the result of the second mathematical expression evaluation to the user interface or script.” (*Id.* 32:18-20.)

– Claim 28, which requires that “the first node is configured to return the result of the second mathematical expression evaluation to the user interface or script.” (*Id.* 33:24-26.)

– Claim 30 includes similar requirements that the third node “communicate the result” and that the first node “return” the result “to a user interface.” (Ex. 1001, 34:42–49.)

Petitioner argues that these limitations are satisfied because, in MultiMATLAB, when `Eval()` is executed on a remote process, its output is visible to the user. (Pet. 17, 45-48.) From that premise, Petitioner infers that a computed result is returned to the initiating node or the user interface. (Pet. 48–49.)

Patent Owner disputes that characterization. The claims do not refer to `Eval()`, and the prior art references do not disclose any return mechanism that

delivers computed values from one node to another for further use, let alone through the claimed path back to the user interface. The analysis that follows addresses these limitations directly and shows that Petitioner's theory rests on inference and generalization, not on what the references actually disclose.

Claim 1, as an example, requires more than a user observing output from a remote computation. It explicitly requires that the third node "communicate the result of the second mathematical expression evaluation to the first node," and that the first node is "configured to return the result of the second mathematical expression evaluation to the user interface." (Ex. 1001, 30:1-3.) Petitioner's theory relies on the proposition that this "return" requirement is satisfied because, in MultiMATLAB, output from remote computations is displayed to the user—i.e., "sent to the user's screen." But those are not the same thing. As demonstrated in the following subsections, a system that forwards screen output for monitoring does not thereby disclose a mechanism that returns the evaluated value to the originating process in a form the process can use, and then returns that result through the system to the user interface as part of the programmed operation.

That mismatch is apparent from the very passages Petitioner cites in the references. Petitioner relies on Menon's statement that the user "interacts directly with" an "interactive process," (Pet. 17) and on Trefethen's description that results

may be “sent to the user’s screen” (*id.* 48; Ex. 1006, 7). Those disclosures describe an interactive front end and a display/monitoring behavior. They do not describe any mechanism by which the *value* computed on a remote node is returned to the originating process as data for further use—much less a system configured to return the result of a second evaluation back to the first node and then to the user interface as claim 1 requires.

Petitioner’s expert fills this hole with innuendo rather than evidence. The Board should reject that effort. *See TQ Delta*, 942 F.3d at 1359–60 (conclusory expert testimony is not substantial evidence); *Magnum Oil Tools*, 829 F.3d at 1380–81 (petitioner’s burden cannot be met with unsupported conclusory statements).

1. The Claims Require Programmatic Return of Evaluation Results

The claim language at issue is directed to a system-level, programmed return path for evaluation results—communication from the executing node back to the originating node and then return to the user interface as part of the cluster’s operation. It is not satisfied by a showing that a remote node prints something that may be visible on a screen.

The specification confirms that understanding. In the embodiment illustrated by Figure 2, the patent explains that results of mathematical expression evaluations executed by the kernel modules “are communicated back to the first cluster module

... via the cluster node modules ... which communicates them to the user interface module.” (Ex. 1001, 6:26–29; Paper 16 at 7.) That disclosure describes an integrated, programmatic return mechanism mediated by the cluster node modules and the user interface module. The patent repeatedly distinguishes this peer-to-peer cluster computing architecture—where nodes exchange information during computation and results flow back through the system’s modules—from systems that merely run computations independently and do not exchange information through a peer-to-peer architecture during execution.

The ’034 Patent claims an architecture in which evaluation results are returned through the system’s modules to the user interface in a manner consistent with the cluster’s programmed operation. That distinction is dispositive for Petitioner’s mapping of limitation [1.6]. If the asserted references do not disclose a mechanism for returning evaluation results to the initiating process (and ultimately to the user interface module) as part of the system’s programmed operation, Petitioner cannot satisfy [1.6] by pointing to screen output alone—particularly where the prior art itself treats screen output as a limited, user-facing display mode distinct from programmatic data return.

2. Trefethen's Eval () Command Does Not Return Results But Rather Forwards Screen Output Only

Trefethen describes Eval () at a user-command level, explaining what the user sees when Eval (' ID ') is run: the remote processes print output (e.g., “ans = 0,” “ans = 1,” etc.), and “the processors are not synchronized and output is sent to the master process as soon as it is ready.” (Ex. 1006, 4.)

Trefethen is explicit that, as described, the system does not return data values from a remote process to the master in response to Eval () . In explaining the state of inter-process communication, Trefethen distinguishes between the Put and Get commands on one hand and the Eval () command on the other and recognizes that with respect to Eval () , “we have not communicated between processes except to send screen output to the master process.” (Ex. 1006, 5.) That sentence is decisive. It draws the exact line Petitioner tries to erase: screen output is forwarded; data is not.

Trefethen's omission of any description of a return-value semantic for Eval () is not an accident. Trefethen's “Implementation of MultiMATLAB” section discusses several commands and their correspondence to MPI operations (e.g., Send, Recv, Bcast, and specialized collectives such as Sum), yet it contains no description of how Eval () is implemented or how it moves or returns data. (Ex.

1006, 11–12.) Trefethen fails to teach how return values are handled as required by the claim limitation.

3. Data Retrieval Requires Separate Get/Send Operations

Trefethen’s architecture confirms that actual data movement is handled by separate commands, not by `Eval()` returning values. Trefethen describes `Put` and `Get` as explicit data transfer mechanisms and emphasizes that they are “executable solely by the master MATLAB process.” (Ex. 1006, 5.) That statement forecloses Petitioner’s implication that a remote process can spontaneously “return” local variables to the master as part of `Eval()`—the only disclosed mechanism for moving variables between workspaces requires master participation. The presence of `Get` is also a practical reduction: these commands would be superfluous if `Eval()` automatically returned computed values. They are not superfluous; they are the disclosed way data is retrieved.

Trefethen’s own examples highlight the same point. At page 6 of Trefethen, the master assigns a value, uses `Send` to transmit the data, and then uses `Eval(3, 'Recv')` to cause the remote node to receive and thereby print the values—values that are then forwarded as screen output. That demonstration explicitly shows the difference between (i) printing an output and (ii) programmatic return. The numeric values are displayed; they are not returned into the master’s workspace as usable

variables. If the system were designed to return evaluated results as data, one would expect a `Get` operation or an assignment of the received value into a variable that is then transmitted back. Trefethen does not show that. It shows a display only, which is not claimed.

Petitioner points to Trefethen's discussion of special operations like `Sum()` and argues that because certain collectives "return" values to a designated process, the system must generally support return semantics for evaluation results. (Pet. 38.) That inference fails. `Sum()` (like `Min` and `Max`) is an explicitly defined collective operation—essentially a programmed reduction built on MPI collectives. It is not a general statement about `Eval()` returning values, and Trefethen provides no enabling disclosure that arbitrary `Eval()` calls return computed values to the master. Trefethen fails to teach or explain that an arbitrary evaluation executed by `Eval()` is returned to the master's workspace.

In short, Trefethen only discloses that `Eval()` dispatches commands, any printed console output may be forwarded to the master's display, and any actual data return requires separate, explicit data-transfer commands.

4. Menon Adds No Return-of-Results Mechanism

Menon confirms the same high-level behavior of `Eval()` as a dispatch mechanism, but it does not disclose any return-of-results functionality.

Menon explains that `Eval()` is “a MEX routine” that takes specified MATLAB process IDs and a command string, and “sends the commands over the underlying communication layer to the processes” with those IDs. (Ex. 1005, 4.) That statement is exactly what it says: commands go out. Menon is silent about any return value, feedback channel, or mechanism for capturing the result of a remote evaluation in the initiating process’s environment. Menon’s discussion is entirely consistent with Trefethen’s user-level view of dispatching commands, observing output, and using separate mechanisms for data movement.

Menon provides no implementation details, and no algorithm, that would support Petitioner’s reading on the claimed requirements. There is no source code or disclosure of any internal mechanism by which a remote evaluation’s result is delivered back to the master’s workspace as data. Nor does Menon suggest that its `Eval()` behaves differently from Trefethen. To the contrary, Menon cites the earlier MultiMATLAB work of Trefethen. (Ex. 1005, 2).

Thus, Menon cannot fill the evidentiary gap. Petitioner is left with the same deficiency: no prior art reference discloses returning evaluation results, and it seeks to gloss over that deficiency with expert narrative.

Petitioner’s institution-stage mapping for limitation [1.6], [25.7], and [28.8] illustrates the problem. Petitioner relied on Menon’s disclosure that “[t]he user

interacts directly with one MATLAB process, called the interactive process” (Pet. 48; Ex. 1005, 3; *see also* Pet. 77, 80), and on Trefethen’s statement that mathematical results are “sent to the user’s screen” when monitoring computations graphically (Pet. 48; Ex. 1006, 7; *see also* Pet. 77,80), and then asserted that when a remote process doubles a value, “MATLAB sends the result to the interactive processor ... to provide the result to the user.” (Paper 16 at 33-34(summarizing Petition).) But that latter point does not appear in the references. It is not a prior art disclosure, but rather an unsupported conclusion.

The cited Trefethen passage about results “sent to the user’s screen” is, on its face, about display/monitoring—not about programmatic return of values through a system architecture. The ’034 Patent’s claims, in contrast, are anchored to a user interface module and a configured return path of evaluation results through the system’s modules. (Ex. 1001, 33:1–5, 33:11–22, 33:23–25; 34:27–31, 34:46–49.) A screen print is not a returned result in the claim sense. The question is not whether a user could observe progress textually or graphically. It is whether the system is configured to return the result of the second evaluation to the user interface, consistent with the claimed architecture. Neither Menon nor Trefethen discloses a mechanism by which the value computed at a remote node is delivered back as data

to the initiating process and then returned to a user interface module as the claims require.

Because the record shows only that output may be displayed, not that evaluation results are returned as data to the user interface module, Petitioner has not carried its burden on [1.6] and related return semantics. *See Arendi S.A.R.L.*, 832 F.3d at 1362–63 (common sense cannot be used to fill a missing limitation where the prior art is silent).

5. Dr. Bajaj’s Eval-Return Assertions Are Conclusory and Unsupported

Petitioner relies on Dr. Bajaj’s declaration to supply the missing disclosure in the references, based on his assertion that MultiMATLAB commands cause worker nodes to “send the result of the computation to the interactive processor.” (Pet. 48, [1.6] referring back to [1.5.2] and Ex. 1003 ¶ 221.) But his assertions are not supported by disclosure in the references and contradict Trefethen’s express statements about what was communicated (“except to send screen output to the master process”) and about master-only data transfer (“puts and gets, executable solely by the master MATLAB process”). (Ex. 1006, 6.) Dr. Bajaj cannot supply missing disclosures by asserting what must be happening “under the hood,” particularly where the reference itself states that the relevant details are not disclosed and where the reference affirmatively distinguishes screen output from data transfer.

The Federal Circuit has repeatedly held that conclusory expert testimony cannot constitute substantial evidence and cannot carry a petitioner’s burden. *TQ Delta*, 942 F.3d at 1359–60; *Magnum Oil Tools*, 829 F.3d at 1380–81. The Board’s rules similarly provide that expert testimony not supported by underlying facts or data is entitled to little or no weight. 37 C.F.R. § 42.65(a). Here, Trefethen and Menon do not disclose the asserted return semantics. Dr. Bajaj’s narrative is therefore mere *ipse dixit*, and it cannot cure the absence of a disclosed return-of-results mechanism.

Accordingly, Petitioner’s “Eval-return” theory fails. The asserted prior art does not disclose programmatic return of evaluation results in the manner required by the claims. Petitioner’s effort to equate screen output with returned results rewrites both the claim language and the prior art, and its reliance on expert assertion to bridge the gap is legally insufficient. Because Petitioner’s mapping of the return and chained-evaluation limitations depends on this flawed premise, the instituted grounds fail for this independent reason as well.

C. The Asserted Prior Art Does Not Teach The Claimed Three-Node Execution Chain

As used in this section, “three-node execution chain” refers to a set of specific claim limitations that require a structured, ordered flow of evaluation and communication across three distinct nodes. Petitioner does not use this label

explicitly, but its mapping theory depends on showing that the prior art teaches a sequence in which:

- a first node sends an evaluation task to a second node;
- the second node performs that evaluation, communicates the result to a third node;
- the third node performs a second evaluation using that result, and
- the third node returns the second result to the first node for delivery to a user interface.

This required sequence is not a general architectural aspiration—it is recited in the actual claim language. For example:

- Claim 1 requires that the second node “execute at least a first mathematical expression evaluation,” that the third node “receive the result” and “execute at least a second mathematical expression evaluation using the result,” and that the third node “communicate the result ... to the first node,” which in turn is “configured to return the result ... to the user interface.” (Ex. 1001, 29:52–30:3.)

- Claims 25, 28, and 30 contain structurally similar language requiring this same multi-node ordered flow. (Ex. 1001, 32:4-17; 33:9-22; 34:39–49.)

When this section refers to “three-node execution chain,” it refers to those specific limitations. Patent Owner uses that phrase as shorthand to reference what

the claims actually require—not as a gloss or generalization. As explained below, Petitioner’s cited references do not disclose a system configured to perform that claimed order of operations. Their attempt to satisfy the claims depends instead on inference, user-authored code, and “could have” speculation, none of which meets the evidentiary standard.

The ’034 Patent does not merely claim that multiple computers can run computations and exchange data. It claims a multi-node evaluation workflow in which intermediate results are communicated across nodes and then returned to the originating node and user interface. Petitioner’s instituted theory attempts to satisfy these limitations by pointing to a single illustrative script in Trefethen (the “cycle”/doubling example), layered atop Menon’s high-level description of an interactive process and MPI point-to-point communication. However, the references do not disclose the claimed system configured to perform the claimed order of operations, and Petitioner’s effort to bridge the gap rests on “could have” reasoning and expert narrative, not on evidence of disclosure in the references.

1. The Claims Require a Specific 1→2→3→1 Order of Operations

Claim 1, as an example, requires, in sequence, that: (i) a second node receives calls from the first node, executes at least a first mathematical expression evaluation, and communicates a result of that first evaluation to a third node; (ii) the third node

receives the result from the second node, executes at least a second mathematical expression evaluation using the received result, and communicates the result of that second evaluation to the first node; and (iii) the first node returns that result to the user interface. (Ex. 1001, 29:52–30:3.)

The architecture described in the specification makes clear why this matters. The patent contemplates a peer-to-peer cluster computing system in which cluster node modules support intercommunication among kernels while tasks execute, and evaluation results are communicated back through the cluster node modules to the user interface module. The claims capture that concept in a concrete ordered operational flow. Petitioner cannot satisfy this limitation by showing that any two nodes can exchange data; it must show the asserted art discloses the claimed evaluation order and result propagation path.

2. No Reference Discloses or Suggests This Architecture

Menon describes a MultiMATLAB architecture in which a user interacts with an “interactive process” and other processes wait for commands from that interactive process. (Ex. 1005, 3, 6.) That is a master-controlled dispatch model. It does not disclose that a second node is configured to send an intermediate evaluation result to a third node as part of an evaluation chain, much less that the third node is

configured to execute a second evaluation using that received result and return the result to the first node for delivery to the user interface.

Trefethen likewise does not disclose the claimed architecture. Petitioner points to an annotated “cycle” or doubling script as purported evidence of the claimed $1 \rightarrow 2 \rightarrow 3 \rightarrow 1$ evaluation chain. (*see, e.g.*, Pet. 37-38, 40-42, 44-47; Ex. 1006, 5–6.) But that script is simply an example of user-authored application code that passes a value from process to process using hardcoded send/receive instructions. It is not a disclosure that the MultiMATLAB system itself is configured to perform a multi-node, evaluation sequence as claimed. The script does not show a first node issuing an evaluation request to a second node, the second node forwarding an intermediate result to a third node for a second evaluation, and the third node returning that result to the first node’s user interface, as the claims require. (Ex. 1001, 29:52–30:4.) Instead, it depicts a prewritten program in which each process performs a fixed action—doubling an input—based solely on its ID. The fact that the behavior is fully scripted only underscores the failure of the Trefethen to disclose the claimed order of operations. Trefethen confirms that the system does not transmit evaluation tasks or results dynamically at runtime. The claims, however, are not directed to what a user might hardcode in a custom script. The claims require the kernels and cluster nodes to execute mathematical evaluations and communicate results across

nodes and back to the originating interface as part of the system's operation. (Ex. 1001, 29:52–30:4.) Petitioner has not shown that the cited art discloses such an architecture.

The claim requires two distinct mathematical expression evaluations executed at two different nodes, with the first evaluation result being communicated to the third node and used in the second evaluation. (Ex. 1001, 29:52–30:4.) The Trefethen script is not described as executing a first expression evaluation at node 2 and a second expression evaluation at node 3 using a received result as claimed. It is a single programmed routine that each node executes, with each node performing the same doubling operation on a transmitted variable. (Ex. 1006, 5–6.) Petitioner's theory therefore improperly requires the Board to treat any manipulation of a received variable as “a first mathematical expression evaluation” and any subsequent manipulation at another node as “a second mathematical expression evaluation,” and then to treat the script's explicit send/receive steps as the claimed communications.

However, neither Menon nor Trefethen discloses the terminal return path required by the claim: the third node communicating the result of the second evaluation to the first node, and the first node returning it to the user interface. Petitioner's attempt to treat “output shown to the user” as satisfaction of that return path fails for the reasons explained in Section IV.B.3, *supra*. The references at most

teach that textual output may be forwarded to a console. They do not disclose programmatic return of evaluation results to the originating node for delivery to a user interface module.

3. Petitioner’s “Could Have” Scenarios Are Legally Insufficient

Petitioner’s fallback position is effectively: because MPI supports point-to-point communication, a POSITA could have arranged for node 2 to send its intermediate result to node 3, and node 3 to send the final result to node 1, thereby achieving the claimed order of operations. (Pet. 44–48 (citing Ex. 1005, 9; Ex. 1006, 6 (annotated)); Ex. 1003 ¶¶ 211–225.) That argument is precisely the kind of “could have” reasoning that fails under controlling law. Obviousness requires proof that the prior art teaches or suggests the claimed arrangement, and that a POSITA would have been motivated to combine the teachings with a reasonable expectation of success. It is not enough to show that the prior art system could be modified in hindsight to behave like the claim. *ActiveVideo Networks, Inc. v. Verizon Commc’ns, Inc.*, 694 F.3d 1312, 1327–28 (Fed. Cir. 2012); *In re Lee*, 277 F.3d 1338, 1343–45 (Fed. Cir. 2002) (conclusory statements are insufficient to support obviousness).

The Federal Circuit has repeatedly rejected attempts to fill missing limitations with conjecture, common sense, or expert narrative where the references are silent.

Arendi S.A.R.L., 832 F.3d at 1362–63 (improper to use “common sense” to supply a missing claim limitation where the prior art is silent or the gap is not plainly filled); *In re Lee*, 277 F.3d at 1343–45 (reversing where PTO relied on conclusory statements and unsupported assumptions rather than articulated reasoning with record support). Where the claim requires a specific ordered sequence and communication architecture, and the cited references do not disclose it, the Board cannot supply it by inference. Petitioner must point to disclosures in the art—not *post hoc* reconstructions—that show or suggest the claimed three-node chain. The absence of any description of nodes configured for this ordered sequence architecture is fatal, particularly given that Menon’s described architecture is master-controlled and not peer-to-peer.

This deficiency is compounded by the fact that Petitioner’s theory depends on treating a user-scripted demonstration as if it were a system-level feature. Even if a user could, with enough programming effort, write an application that passes values along a chain of processes, that does not establish that the underlying system disclosed in the references is configured for the claimed ordered sequence architecture. The gap between what is theoretically possible with MPI and what is actually disclosed in the references cannot be bridged by expert say-so without violating the rule against impermissible gap-filling.

4. Consistency with Prior PTAB Decisions Rejecting Similar Theories

This Board has already addressed materially similar “third-node chain” theories in prior ACS proceedings. In the earlier NVIDIA challenges to related ACS patents, the Board rejected arguments that relied on speculative “may/could” scenarios to satisfy a three-node chained evaluation limitation. The Board found that such expert testimony lacked record support and was entitled to little weight where the cited reference did not disclose the claimed third-node behavior. (IPR2021-00019, Paper 9, 28, 30; IPR2021-00020, Paper 9, 27, 29.) The same principle applies here: Petitioner’s chained-evaluation theory relies on relabeling and inference rather than disclosure, and the cited art does not teach the required configuration.

While each case is decided on its own record, the logic is the same and the evidentiary defect is the same. Where the claim requires a specific multi-node chained order of operations, and the cited references do not disclose it, a petitioner cannot prevail by hypothesizing how a skilled programmer could write additional code to approximate the claimed behavior. That is exactly the gap-filling the Federal Circuit and the Board do not permit.

For these reasons, Petitioner has not carried its burden to show that the asserted prior art teaches or suggests the claimed three-node execution chain. This is an independent, dispositive defect in Petitioner’s obviousness case.

V. NO MOTIVATION TO COMBINE AND NO REASONABLE EXPECTATION OF SUCCESS

Petitioner bears the burden to prove that a POSITA would have been motivated to combine the asserted references to achieve the claimed invention and would have had a reasonable expectation of success in doing so. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007); *Procter & Gamble Co. v. Teva Pharms. USA, Inc.*, 566 F.3d 989, 994 (Fed. Cir. 2009). As shown herein, Petitioner has failed to meet this burden.

A. Petitioner Offers Only a General Goal, Not a Record-Based Motivation to Build the Claimed System

Petitioner's motivation-to-combine Trefethen with Menon argument reduces to the unremarkable proposition that researchers wanted to parallelize high-level mathematical computing environments. Paper 16, 21 (quoting Petitioner's motivation: "because researchers had a strong desire to parallelize MATLAB"). That generalized desire does not supply a reason to combine these references in the specific manner required by the claims, nor does it explain why a POSITA would have selected and implemented the particular multicore, result-return, and ordered sequence architecture recited in the claims.

Motivation must be tied to the claimed configuration, not to an abstract goal. *See KSR*, 550 U.S. at 418 (obviousness requires an articulated reason to combine to

achieve the claimed invention, not hindsight). Here, Petitioner's asserted reasons to combine do not explain how or why a POSITA would arrive at the claimed inventions, as opposed to any number of other master-controlled, node-parallel arrangements that were commonplace at the time.

B. Menon Is A Redesign, Not An Extension, Of Trefethen

Petitioner's proposed combination treats Menon as if it were an incremental extension of Trefethen that a POSITA could merge into a predictable, unified MultiMATLAB system. (Pet. 22 (asserting Menon "builds on" the MultiMATLAB system taught by Trefethen); Pet. 22–23 (asserting Menon and Trefethen "together represent a combination of prior art elements according to known methods ... to yield the predictable result" of MultiMATLAB executing MATLAB in parallel).) The Menon paper itself refutes that premise. Menon expressly 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. (Ex. 1005, 14 (stating "The first MultiMATLAB architecture is different from the newer one described in this paper" and noting it was "more tightly coupled to MPICH... resulting in some performance limitations" and "does not permit extensions via parallel MEX routines").) Thus, Menon itself explains it is an architectural replacement of Trefethen. The authors plainly did not view the original

architecture as a stable platform for layering on capabilities; they viewed it as limited and therefore built a different architecture.

This matters because Petitioner’s asserted combination assumes compatibility and predictability: that one can take Trefethen’s architecture and simply layer Menon’s improvements on top, plus RS6000 hardware details and POE initialization, to arrive at the claimed peer-to-peer, ordered sequence evaluation system with multicore nodes. (Pet. 22–23 (asserting Menon and Trefethen “together represent a combination of prior art elements according to known methods ... to yield the predictable result” of MultiMATLAB executing MATLAB in parallel); Pet. 24–25 (asserting motivation to combine RS6000 with Menon because Menon uses the IBM SP2/RS6000 SP system); Pet. 26–27 (asserting motivation to combine POEref with Menon because Menon teaches MultiMATLAB uses POE).) That is not a “combination of prior art elements according to known methods.” Rather, Petitioner attempts to use the claims as a blueprint to redesign the prior art—precisely what obviousness law forbids.

C. Menon Identifies Fundamental Deficiencies In The Earlier Architecture

Menon explains why the earlier architecture of Trefethen was deficient. Menon states that the first architecture was more tightly coupled to a specific MPI implementation, resulting in performance limitations, and—critically—that it “does

not permit extensions via parallel MEX routines.” (Ex. 1005, 14.) Menon identifies structural limits that prevented the first architecture (Trefethen) from supporting the kinds of extensions and performance that users wanted.

Petitioner’s theory requires the Board to assume that, notwithstanding Menon’s express critique of Trefethen, a POSITA would have considered it straightforward to merge or augment the older system to achieve the claimed invention. That is implausible on the record. The Menon authors who knew the first system best did not describe a path of incremental improvement; they described the need for a different architecture entirely. When the prior art itself identifies fundamental constraints that prevented extension, it undermines any claim that a POSITA would have been motivated to take the very path the art implies is not viable. *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 567 F.3d 1314, 1326–27 (Fed. Cir. 2009) (A reference that teaches away weighs against a finding of obviousness.). This is not simply relevant to teaching away (addressed separately); it also defeats Petitioner’s motivation and expectation showing. Menon says “the old design cannot support the extension,” thus Petitioner cannot simultaneously claim that combining the old design (Trefethen) with other references would predictably yield the claimed architecture. This point is fatal to Petitioner’s proposed combination.

D. The Proposed Combination Would Not Yield The Claimed System

Even accepting Petitioner’s “desire to parallelize MATLAB” narrative, the proposed combination still would not yield the claimed system without additional inventive steps. Several independent problems remain unsolved by the cited references.

First, Petitioner’s combination does not disclose a multi-core hardware processor as required by the claims. RS6000 describes SMP nodes populated with multiple discrete processors, and does not disclose a processor that comprises multiple cores as required by all claims. Menon and Trefethen likewise operate in the single-core, multi-processor world of the 1990s. That is not a small gap. It is a missing limitation in every independent claim.

Second, the cited references do not disclose the programmatic return of evaluation results and the three-node ordered sequence evaluation required by the claims. As explained in Section IV.B., Trefethen’s disclosure distinguishes screen output forwarding from data transfer, and the references do not describe any mechanism by which evaluated results are returned to the originating node’s user interface module as the claims require.

Third, the asserted references do not disclose the claimed ordered operational flow. Menon explicitly describes an interactive process from which other processes

“wait for commands.” That is a master-controlled dispatch architecture, not a peer-to-peer architecture for task delegation at the node level. Petitioner cannot transform that master-controlled architecture into the claimed peer-to-peer architecture merely by pointing out that MPI supports point-to-point data transfers. The question is not whether MPI could transmit data between any pair of nodes. The references must disclose a system configured to perform the claimed ordered sequence evaluation and return a result through the nodes to the user interface module. Menon does not.

When a proposed combination would still lack multiple required elements, Petitioner cannot demonstrate reasonable expectation of success. Reasonable expectation of success is a distinct requirement; it cannot be presumed where the proposed combination still lacks multiple required elements. *In re Cyclobenzaprine Hydrochloride Extended-Release Capsule Patent Litig.*, 676 F.3d 1063, 1070–71 (Fed. Cir. 2012) (reasonable expectation of success is a separate, fact-based requirement and is not satisfied by a generalized motivation to try or by hindsight reconstruction). A POSITA cannot reasonably expect success in achieving the claimed invention from the combination Petitioner proposes when the combination, as disclosed, does not provide the necessary hardware architecture or the required result-return and ordered sequence evaluation functionality. Petitioner’s argument collapses into “a POSITA could have invented the missing parts,” which is not a

substitute for proof that the claimed invention would have been obvious based on the teachings of the prior art references.

E. Petitioner's Theory Relies on Impermissible Hindsight

Petitioner's motivation and expectation narrative is built on hindsight. It starts with the claim requirements and then constructs a pathway through disparate statements in the references to say the system could be made to behave that way. But the contemporaneous record does not show that pathway. It shows the opposite: Trefethen's first-generation MultiMATLAB had architectural limitations; Menon redesigned it; the claimed features remain absent from both; and Petitioner then impermissibly fills those disclosure gaps with using the claims as the guide.

The Federal Circuit has repeatedly cautioned that obviousness analysis cannot be driven by the inventor's disclosure. *Intercontinental Great Brands LLC v. Kellogg N. Am. Co.*, 869 F.3d 1336, 1349–50 (Fed. Cir. 2017) (improper to use hindsight and treat the patent as a roadmap for selecting and combining prior art); *In re Dembiczak*, 175 F.3d 994, 999 (Fed. Cir. 1999) (warning against hindsight reconstruction using the inventor's disclosure as a blueprint). A petitioner must identify record-based reasons why a skilled artisan would have been led to the claimed invention and would have had a reasonable expectation of success. Here, Petitioner's asserted motivation is essentially a generalized desire to parallelize high-

level computing environments. (Pet. 23–24 (asserting “researchers had a strong desire to parallelize MATLAB” as motivation to combine Menon and Trefethen).) That desire does not supply a reason to combine these particular references in this particular way to achieve the claimed architecture, especially where the references themselves criticize the earlier design and do not disclose the missing elements. Nor does it supply a reasonable expectation that the combination would succeed in producing the claimed multicore peer-to-peer ordered sequence evaluation system. Petitioner’s theory therefore rests on impermissible hindsight, and the Board should reject it on that basis as an independent ground for denying unpatentability.

For these reasons, Petitioner has not met its burden to establish motivation to combine or reasonable expectation of success for any instituted ground.

VI. MENON TEACHES AWAY FROM PETITIONER’S PROPOSED COMBINATIONS

Petitioner’s combination theory is undermined by the prior art itself. A reference teaches away when it criticizes, discredits, or otherwise discourages the path Petitioner proposes such that a POSITA would be led in a different direction. *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir. 1994); *DePuy Spine*, 567 F.3d at 1326. That is precisely the situation here. 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. (Ex.

1005, 14 (stating “the first MultiMATLAB architecture is different from the newer one,” was “more tightly coupled to MPICH ... resulting in some performance limitations,” could not use vendor-optimized MPI implementations, and “does not permit extensions via parallel MEX routines”).) Petitioner’s theory nevertheless treats Trefethen’s first-generation system as a foundational building block and urges the Board to accept that one would have simply combined Trefethen’s earlier architecture with Menon’s later work (and other IBM materials) to reach the claimed invention. Menon teaches away from that combination.

A. Express Criticism of the First-Generation MultiMATLAB Architecture

Menon is explicit: “The first MultiMATLAB architecture is different from the newer one described in this paper.” (Ex. 1005, 14.) Menon then explains why. It states that the first system was tightly coupled to a particular MPI implementation, resulting in performance limitations, and that it could not take advantage of vendor-optimized MPI implementations. *Id.* Menon further identifies that the earlier architecture was not designed to support later, desired extensions. *Id.* Menon expressly states that Trefethen’s first-generation architecture is inadequate for the objectives that motivated the newer design.

Plainly this is not an invitation to combine first- and second-generation architectures. The earlier approach had structural constraints that prompted the

Menon authors to move away from it. Where the authors of a reference describe an earlier system as limited and emphasize that their newer system is architecturally different to address those limitations, the reference is discouraging reliance on the earlier system as a platform for the very improvements the newer system enables. *DePuy Spine*, 567 F.3d at 1326–27 (teaching away where a POSITA would be discouraged from the proposed approach, undermining obviousness); *In re Gurley*, 27 F.3d at 553 (teaching away when the reference suggests the proposed path is undesirable or unlikely to work). That is teaching away in the straightforward sense: the reference tells the skilled artisan that the earlier path is not the path to the desired result.

Petitioner’s combination theory depends on the opposite premise: that the earlier architecture is a viable foundation on which to build, and that a POSITA would naturally rely upon it with other disclosures to reach a peer-to-peer, ordered sequence evaluation system with multicore hardware. Menon’s critique makes that premise untenable. If Trefethen’s first system was architecturally constrained in the ways Menon describes, a POSITA would understand that the “obvious” direction was to adopt the newer architecture and abandon the earlier one.

B. Incompatibility With Parallel MEX Extensions

Menon's most pointed teaching-away statement is directed to extensibility: the first MultiMATLAB architecture "does not permit extensions via parallel MEX routines." (Ex. 1005, 14.) It is difficult to imagine clearer teaching away language in a technical paper. The Menon authors plainly state that the earlier Trefethen system is not compatible with a class of extensions that were important enough to be featured as a primary advantage of the newer system.

Petitioner's obviousness story is built around combining references to supply missing functionality. Petitioner's approach requires the Board to assume that, even if the earlier system did not originally have certain capabilities, a POSITA would have found it straightforward to add them using later-developed concepts and tools. Menon says the opposite: the architecture itself does not permit such extensions. When a reference teaches that a specific approach cannot support a desired extension, it discourages a POSITA from pursuing that approach to achieve that extension. *DePuy Spine*, 567 F.3d at 1326.

Petitioner may argue that Menon's statement is limited to parallel MEX routines and not to other features. But that argument misses the point. Parallel MEX extensibility is but one concrete example Menon gives that the first architecture was not a reliable foundational architecture.

C. Teaching Away Undermines Any Motivation to Combine

Teaching away is not merely an additional argument; it directly undermines Petitioner's required motivation-to-combine showing. An obviousness theory premised on combining references must be supported by a reason why a POSITA would have pursued the combination with a reasonable expectation of success. Where the prior art discourages the path, that reason is missing. *In re Gurley*, 27 F.3d at 553. Here, 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.

This teaching away is especially significant in this case because the references at issue are not unrelated; they are sequential works by related authors on the same project. The authors had every incentive to reuse their earlier work if a straightforward upgrade path existed. Instead, Menon describes the new architecture as different and identifies limitations that the earlier architecture could not overcome. That contemporaneous evidence—coming from the project's own developers—strongly supports Patent Owner's position that the path Petitioner proposes is not the path the prior art taught. It also reinforces the lack of reasonable expectation of success discussed in Section V: when the designers of the system

themselves state that certain extensions are not permitted by the first architecture, it is not credible for Petitioner to argue that a POSITA would have expected to achieve the patent's more demanding architectural requirements by combining the earlier and later references.

Accordingly, Menon's teaching away provides an independent and compelling reason to reject Petitioner's proposed combinations.

VII. PETITIONER'S EXPERT TESTIMONY CANNOT CURE MISSING CLAIM ELEMENTS

Petitioner's instituted grounds depend on using expert narrative to supply what the asserted references do not disclose. That approach is legally insufficient. The Board's final written decision must rest on what the prior art actually teaches and on a supported rationale for combination—not on conclusory expert assertions that rewrite the references or fill gaps by hypothesis. In this proceeding, Petitioner bears the burden of persuasion throughout. 35 U.S.C. § 316(e); *Magnum Oil Tools*, 829 F.3d at 1375. Petitioner cannot meet that burden by asking the Board to accept Dr. Bajaj's conclusions in lieu of documentary proof of claim limitations. Where a claim limitation is missing from the asserted references, expert testimony cannot manufacture it. *Arendi S.A.R.L.*, 832 F.3d at 1362–63 (common sense and expert intuition cannot fill gaps where the prior art is silent).

This proceeding illustrates those principles in a stark way. Petitioner relies on expert *ipse dixit* for at least two critical claim requirements: (i) treating RS6000's SMP node configurations as disclosing a multi-core processor, and (ii) attributing to MultiMATLAB an "Eval-return" and three-node ordered sequence evaluation behavior not disclosed by Trefethen or Menon. Those assertions are either unsupported by the references or contradicted by them. They cannot carry Petitioner's burden.

A. Conclusory Assertions Are Not Substantial Evidence

An expert's opinion is not self-proving. PTAB decisions must be supported by substantial evidence, and conclusory expert testimony—unsupported by underlying facts in the cited references—is not substantial evidence. *TQ Delta*, 942 F.3d at 1359–60. The Board's rules reflect the same principle: "[e]xpert testimony that does not disclose the underlying facts or data on which the opinion is based is entitled to little or no weight." 37 C.F.R. § 42.65(a).

Here, Dr. Bajaj repeatedly supplies critical claim elements by mere unsupported assertion. For example, he asserts that RS6000's "SMP" is the claimed "hardware processor," and that RS6000's multiple discrete processors per node are "processing cores," such that "each node has a hardware processor with a plurality of processing cores." (Ex. 1003 ¶¶ 207–210.) But RS6000 does not say that. It

describes a SMP “configuration” and says “processors per node,” (Ex. 1007, 2, 9.) The expert’s relabeling is not a disclosure. Likewise, Dr. Bajaj asserts that Trefethen’s MultiMATLAB returns the result of an evaluation to the interactive process and ultimately to the user interface (Ex. 1003 ¶¶ 226–228), when Trefethen expressly distinguishes screen output forwarding from data transfer and identifies separate mechanisms (e.g., master-only put/get) for actual data movement. (Ex. 1006, 5.) These are not reasonable inferences drawn from disclosed facts; they are conclusory statements offered to patch evidentiary holes.

Federal Circuit precedent does not allow such patching. *Magnum Oil Tools* makes clear that a petitioner cannot satisfy its burden with conclusory assertions; the petitioner must affirmatively prove unpatentability with evidence and reasoned explanation. 829 F.3d at 1380–81. Petitioner’s reliance on expert *ipse dixit* is thus legally inadequate. Where the references are silent or inconsistent with the expert’s narrative, Petitioner has not met its burden.

B. Conflicts with the Written Record Must Be Resolved Against Petitioner

When expert testimony conflicts with the plain disclosure of a prior art reference, the reference controls. The Board errs by crediting expert testimony that contradicts the reference’s express teachings. *In re Lee*, 277 F.3d at 1343–45. That principle is particularly important in IPR proceedings, where the Board’s task is to

determine what the prior art discloses based on record evidence, not to accept conclusory expert assertions untethered from the cited references. *See also Arendi S.A.R.L.*, 832 F.3d at 1362–63.

Petitioner’s expert-driven theories collide with the written record in multiple places. On the multicore limitation, RS6000’s text repeatedly describes “processors per node,” an SMP “configuration,” and processor cards used to populate nodes with multiple discrete CPUs. Nothing in RS6000 describes a single processor having multiple cores. Dr. Bajaj’s mapping requires the Board to disregard that architecture and to treat discrete CPUs as “cores.” That is inconsistent with both the claim language and the intrinsic record, which distinguishes processors and cores and expressly discusses multiple cores being included in a single-chip package. Where the reference and the patent draw that distinction, expert testimony cannot erase it.

On return evaluation results, Petitioner’s “Eval-return” theory depends on treating console output or user-visible output as a returned evaluation result to the user interface, and on attributing to Eval () an implicit return value mechanism. Yet Trefethen explicitly states that the only inter-process communication implemented (at least at the stage described) was sending screen output to the master, and it separately describes master-only mechanisms for puts/gets as the way data is transferred. Those disclosures are inconsistent with Petitioner’s suggestion that

evaluation results are returned programmatically by default. Where the prior art expressly describes only screen output forwarding and separate data-transfer commands, the Board should not accept an expert's contrary narrative. The prior art disclosure must control.

The Board should therefore resolve such conflicts against Petitioner and reject any expert mapping that depends on contradicting the references.

C. Petitioner Is Bound by the Theories in the Petition

Petitioner must stand or fall on the theories it advanced in the Petition. It may not use reply briefing or post-institution argument to introduce new rationales or new combinations to cure deficiencies in its original case. *Magnum Oil Tools*, 829 F.3d at 1381 (rejecting attempts to shift burden or introduce new arguments to patch the petition); *see also*, 37 C.F.R. § 42.23(b) (reply may respond only to arguments raised in the corresponding opposition/patent owner response).

This principle is critical here because Petitioner's instituted grounds have identifiable, dispositive proof failures. Petitioner's mapping of the multicore limitations relies on RS6000's SMP disclosure; Petitioner's mapping of the return and ordered sequence evaluation limitations relies on Menon's interactive process and Trefethen's scripts/output, supplemented by Dr. Bajaj's characterizations. If those mappings fail on the merits—as they do—Petitioner cannot salvage the case

by pivoting to a new hardware theory, a new source for multicore disclosure, a new interpretation of Eval () returning results, or a new explanation of how the three-node chain is disclosed elsewhere. Those would be new theories improperly expanding the petition. Petitioner's burden is to prove unpatentability based on the instituted case, and that instituted case cannot be repaired by a moving target.

Accordingly, the Board should evaluate Petitioner's grounds based on the asserted evidence and the petition's own mappings. When that is done, the conclusion follows: Petitioner's expert testimony cannot cure missing claim elements, cannot contradict the written record, and cannot substitute for the disclosures that Petitioner was required to prove.

VIII. PETITIONER'S REFERENCES ARE NOT "PRINTED PUBLICATIONS"

Petitioner bears the burden to prove that "Menon" (Ex.1005), "Trefethen" (Ex.1006), "RS/6000" (Ex.1007), "POEref" (Ex.1008), and "MPIref" (Ex.1017) qualify as a "printed publication." To qualify, a reference must be "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." *Kyocera Wireless Corp. v. Int'l Trade Comm'n*, 545 F.3d 1340, 1350 (Fed. Cir. 2008); *In re Lister*, 583 F.3d 1307, 1311 (Fed. Cir. 2009). Competent evidence is required. *Medtronic, Inc. v. Barry*, 891 F.3d 1368, 1380 (Fed. Cir. 2018); *Blue Calypso, LLC*

v. Groupon, Inc., 815 F.3d 1331, 1348–51 (Fed. Cir. 2016). Because Petitioner has not proven public accessibility, none of its references qualifies as prior art, and all grounds fail.

A. IBM Technical Manuals (RS/6000 and POEref, Exs. 1007–1008)

Petitioner asserts that the IBM manuals were indexed in IBM’s online library by 2001–2002 and thus qualify as printed publications. But Petitioner offers no evidence that the manuals were ever publicly accessible. At most, they were hosted on an IBM repository. Petitioner must show that interested researchers could have found them through reasonable diligence. *Samsung Elecs. Co. v. Infobridge Pte. Ltd.*, 929 F.3d 1363, 1369 (Fed. Cir. 2019). Yet Petitioner provides no evidence that a POSITA using typical search methods at the time could have located them.

The Petition cites printouts of IBM webpage “link chains” (Exs.1038, 1039) and an Internet Archive timestamp (Ex.1029) to suggest the manuals were hosted on IBM’s website by 2001. But these printouts do not show how the public could have navigated to the manuals. There is no evidence of any IBM webpage from that time listing the manuals or allowing users to search for them by topic.

Dr. Bajaj did not base his conclusions on any evidence that the manuals were indexed or searchable on IBM’s website, and the record includes none. (Ex. 1003 ¶¶119, 123-124.) At most, the manuals might have been accessible to someone who

knew the exact URL—not “sufficiently accessible to the public interested in the art.” *In re Cronyn*, 890 F.2d 1158, 1161 (Fed. Cir. 1989). Petitioner cites no evidence that anyone ever accessed or even knew of the manuals before the critical date. Their mere presence on an IBM server is like the unindexed thesis in *Cronyn*—public-facing, but undiscoverable. *See also Blue Calypso*, 815 F.3d at 1348.

Petitioner’s evidence shows the manuals were on IBM’s server by 2005, but not that they were searchable via internet engines. Although Petitioner notes the manuals shipped with RS/6000/SP2 systems, distribution to customers or user groups is not public accessibility without evidence they reached non-customers. *Medtronic*, 891 F.3d at 1380. Nothing shows the manuals were disseminated beyond IBM’s customer base or made available in libraries or public forums.

B. MPI Forum Draft Standard (“MPIref,” Ex. 1017)

Petitioner contends that Exhibit 1017 (“MPIref”) was distributed via FTP servers and posted on the MPI Forum’s website in 1994, and thus qualifies as a printed publication. But Petitioner’s evidence does not show actual public accessibility. It cites only language within the draft stating it was “processed by LaTeX on May 4, 1994” and “made available” by the working group—statements in past tense, embedded in the document, and uncorroborated. Such assertions,

without evidence of indexing, searchability, or public dissemination, are insufficient.

In re Klopfenstein, 380 F.3d 1345, 1350 (Fed. Cir. 2004).

Petitioner also relies on Dr. Bajaj, who asserts that the MPI draft was available via an FTP “mail server” and linked from certain websites. See Ex. 1003 ¶¶129-130. But Petitioner offers no evidence that anyone outside the MPI Forum ever knew of or accessed the draft. There is no contemporaneous documentation—no listserv announcement or public notice—and Dr. Bajaj does not claim he or anyone else retrieved it. The cited web link does not yield the document. At most, Petitioner shows possible access by insiders, not public accessibility. Courts and the Board have held that distribution to a closed group or those already aware of the work is insufficient. *Samsung Elecs. Co. v. Infobridge Pte. Ltd.*, 929 F.3d 1363, 1374–75 (Fed. Cir. 2019); *Argentum Pharm. LLC v. Research Corp. Techs.*, IPR2016-00204, Paper 19 at 11–12 (PTAB May 23, 2016).

C. Cornell “MultiMATLAB” Web Article (“Trefethen,” Ex. 1006)

Petitioner alleges that Trefethen (Ex. 1006) was publicly accessible via a Cornell University webpage and that one of the authors linked to it from a personal homepage. (Petition at 15; Ex. 1003 ¶¶116-117.) But Petitioner offers no competent evidence that such a hyperlink existed or that anyone actually found the page in the

1990s. (Ex. 1023.) The record contains no proof of indexing, public discoverability, or that the site was accessed by any member of the public. (Ex. 1029.)

First, the archived evidence contradicts Petitioner's story of an author's webpage hyperlink. The very webpage that was supposed to be the gateway to Trefethen does not actually reference or link to the article. At best, 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.

Second, even if Trefethen resided on a Cornell server in 1996, Petitioner offers no evidence that anyone knew of it or could find it. “Public accessibility requires more than technical accessibility.” *Samsung Elecs. Co. v. Infobridge Pte. Ltd.*, 929 F.3d 1363, 1369 (Fed. Cir. 2019). Petitioner cites no search index, library, or database that cataloged Trefethen, nor any proof that Cornell's computer science pages were indexed or publicly visible in the 1990s. The record contains no evidence that the department or author webpages were discoverable by a POSITA. (Ex. 1029.) Cornell's popularity is not evidence that Trefethen was publicly available.

Third, 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. But posting a document on a personal website does not make it a printed publication

unless the site was publicly known or indexed. *Blue Calypso*, 815 F.3d at 1349 (Fed. Cir. 2016). Petitioner cites no evidence that anyone would have known of Ms. Trefethen’s personal page or located the article on Cornell’s server.

Finally, Petitioner’s claim that Trefethen was “cited 10 times” before the relevant date does not establish public accessibility. (Pet. 15; Ex. 1003 ¶¶116; Ex. 1035 at 2–3.) Citation alone does not prove that the public could locate the document. Indeed, one such citation appears in the Menon paper itself (Ex. 1005 at 1–2)—an instance of self-citation, not evidence of dissemination.

D. SC’97 Conference Paper (“Menon,” Ex. 1005)

Unlike the other references, Exhibit 1005 (“Menon”) was part of a known academic conference. Petitioner argues that Menon “was published and distributed at the Proceedings of the 1997 ACM/IEEE Conference on Supercomputing” and thus is a printed publication. Petition at 14 (citing Ex. 1005; Ex. 1027 ¶ 11.) Petitioner further notes that Menon later became available on Cornell’s website by 2003 and was cited by numerous subsequent papers. *Id.*; Ex. 1003 ¶¶110-112; Ex. 1028.

First, however, Petitioner has not proven that the SC’97 conference proceedings (including Menon) were actually disseminated to the public in 1997. Petitioner relies heavily on the declaration of Gordon MacPherson, an IEEE

publications custodian (Ex. 1027), who stated that copies of the SC’97 proceedings “were made available no later than the last day of the conference.” But MacPherson does not explain how or to whom the proceedings were distributed. Moreover, MacPherson provides no foundation for his knowledge. He does not claim to have been present at SC’97, nor to have personally overseen distribution of the proceedings. He provides no supporting documentation or contemporaneous evidence to corroborate his statement. Aside from his bare assertion, there is nothing in the record to indicate that anyone actually obtained Menon during or shortly after the SC’97 conference.

Notably, MacPherson’s declaration references an attached “Exhibit A”—purportedly a copy of Menon retrieved from IEEE Xplore in March 2025—and claims that the IEEE Xplore entry “show[s] the date of publication.” (Ex. 1027 ¶ 10.) But upon inspection, “Exhibit A” (Ex. 1027 at 3–21) does not display any publication date for Menon. Petitioner did not provide the IEEE Xplore abstract or bibliographic metadata page, so the supposed evidence of publication date is missing from the record. As a result, MacPherson’s assertion that IEEE Xplore confirms a 1997 publication date is uncorroborated.

Apart from IEEE’s internal records, Petitioner suggests that Menon was available through other channels: a Cornell website and the Supercomputing ’97

conference website. (Pet. at 14–15; Ex. 1003 ¶¶110-111.) But Petitioner offers no evidence that the Menon PDF was indexed or that anyone actually located it on Cornell’s site within the relevant timeframe. (Ex. 1029 (archived Cornell webpage lacking indexing metadata or navigable links)). As for the SC’97 website, Petitioner’s expert merely suggested that conference papers could have been accessible there, but submitted no evidence that the site actually hosted full papers for public download or that it was indexed or known to the broader community. (Ex. 1003 ¶110.) Moreover, Exhibits 1030 and 1031 relate to the conference, but do not identify any URL for a SC’97 webpage where Menon was allegedly available. Petitioner has failed to show that any member of the public—who did not attend SC’97—could have found or accessed the Menon paper.

Even if Menon eventually became publicly available, the question is when and how it became available, and whether that meets the legal standard for a printed publication. *See Medtronic*, 891 F.3d at 1380. Petitioner must prove that Menon entered the public domain no later than the critical date and has failed to do so.

E. Petitioner’s Citation-Based Arguments Will Be Addressed Separately

In an attempt to patch the holes in its public accessibility showing, Petitioner points to lists of other publications that cite the Menon, Trefethen, and MPIref references (Exhibits 1028, 1035, and 1037). Petitioner invites the Board to infer that,

because certain papers (published before 2006) have bibliographic citations to these references, the references must have been publicly available. That argument raises both evidentiary and substantive issues, including whether the cited references correspond to the asserted exhibits and whether they were publicly accessible at the relevant time. Patent Owner will address those issues in a forthcoming motion to exclude.

IX. PRESERVED EVIDENTIARY ISSUES

Patent Owner preserves its objections regarding the foundation for Petitioner's public-accessibility evidence. (Ex. 2029 at 2-4.)

X. CONCLUSION

For all these reasons, Petitioner has not met its burden to prove unpatentability by a preponderance of the evidence. The asserted prior art does not disclose or suggest the claimed multicore hardware processor, the programmatic return of evaluation results, or the specific ordered sequence the claims require. Petitioner's reliance on expert testimony to supply these missing elements is legally insufficient, as expert assertions cannot substitute for what the references fail to disclose.

In addition, Petitioner has not established a legally sufficient motivation to combine the asserted references or a reasonable expectation of success, and Menon's own critique of the earlier MultiMATLAB architecture teaches away from

Petitioner's combination theory. Petitioner also fails to carry its burden on the statutory threshold requirement that certain asserted references qualify as printed publications, and Patent Owner's preserved evidentiary objections further undermine the weight that may be accorded to Petitioner's proof.

Accordingly, the Board should determine that Petitioner has not proven unpatentability, and all challenged claims should be confirmed.

Dated: February 6, 2026

Respectfully submitted,

/David P. Lindner/

David P. Lindner, Reg. No. 53,222

Counsel for Patent Owner

Advanced Cluster Systems, Inc.

CERTIFICATE OF TYPE-VOLUME LIMITATIONS

Pursuant to 37 C.F.R. § 42.24(d), Patent Owner certifies that this Patent Owner Response complies with the type-volume limits of 37 C.F.R. § 42.24 because it contains 13,952 words, according to the word-processing system used to prepare this brief, excluding the parts exempted by 37 C.F.R. § 42.24.

Dated: February 6, 2026

Respectfully submitted,

/David P. Lindner/

David P. Lindner, Reg. No. 53,222

Counsel for Patent Owner

Advanced Cluster Systems, Inc.

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. § 42.6(e), the undersigned certifies that I caused a true and correct copy of the foregoing Patent Owner Response to be served in its entirety on February 6, 2026, by filing this document through the U.S. Patent Office's P-TACTS Filing System, as well as causing true and correct copies be delivered by electronic mail on Petitioner's lead and backup counsel at the following email addresses of record:

Andrew S. Ehmke
David L. 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 E. Ferguson
Chaoxuan Charles Liu
James Kappos
WINSTON & STRAWN LLP
beferguson@winston.com
AMD-IPR@winston.com
ccliu@winston.com
jkappos@winston.com

IPR2025-00913 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,570,034

Dated: February 6, 2026

Respectfully submitted,

/David P. Lindner/

David P. Lindner, Reg. No. 53,222

Counsel for Patent Owner

Advanced Cluster Systems, Inc.