

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

INTEL CORPORATION,

Petitioner,

v.

ADVANCED CLUSTER SYSTEMS, INC.,

Patent Owner.

IPR2025-00916

U.S. Patent No. 11,811,582

PATENT OWNER RESPONSE

TABLE OF CONTENTS

I.	Introduction.....	1
A.	The '582 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	7
II.	Procedural Background and Governing Legal Standards	8
A.	Instituted Grounds and Challenged Claims.....	8
B.	Petitioner's Burden of Proof Never Shifts	8
C.	Limits on Expert Testimony and Impermissible Gap-Filling	8
D.	Reliance on the Existing Evidentiary Record	9
III.	Claim Construction.....	9
A.	Governing Claim Construction Principles	9
IV.	Petitioner Fails to Establish a <i>Prima Facie</i> Case of Obviousness	10
A.	Petitioner Fails to Establish the Claimed Return of Evaluation Results Required by Claim 1	11
1.	Claim 1 Requires Programmatic Return of Evaluation Results, Not Mere Execution or Display	11
2.	Menon and Trefethen Do Not Disclose Programmatic Return of Evaluation Results	12
3.	Screen Output Is Not a "Returned Result" Under Claim 1	13
4.	Petitioner's Eval-Return Theory Is Based on Expert <i>Ipse Dixit</i> , Not Disclosure.....	14
5.	The Failure to Disclose Result Return Is Dispositive	15
B.	Petitioner Fails to Establish the Claimed Multi-Stage, Dependent Evaluation Flow Required by Claim 1	15
1.	Claim 1 Requires a Specific, Dependent Multi-Stage Evaluation Using Returned Results	15
2.	Menon and Trefethen Do Not Disclose Kernels Using Other Kernels' Results as Inputs to Further Computation.....	16

3.	Trefethen's cycle.m Example Does Not Disclose the Claimed Evaluation Chain.....	18
C.	Petitioner Fails to Establish the Claimed Cluster Configuration Module Required by Claim 4.....	19
1.	Claim 4 Requires an Internal Cluster Configuration Module That Initializes Cluster Node Modules	19
2.	Menon Does Not Disclose Any Cluster Configuration Module ..	20
3.	IBM's Parallel Operating Environment Is an External Job-Launch Tool, Not a Cluster Configuration Module.....	21
4.	Job Launching Is Not the Same as Initializing Cluster Node Modules.....	22
5.	Petitioner's Theory Relies on Impermissible Gap-Filling and Expert <i>Ipse Dixit</i>	22
6.	The Failure to Disclose a Cluster Configuration Module Independently Defeats Claim 4.....	23
V.	No Motivation to Combine and No Reasonable Expectation of Success	23
A.	Petitioner Offers Only A General Goal, Not A Record-Based Motivation to Build the Claimed System.....	24
B.	Menon Is A Redesign, Not An Extension, Of Trefethen	25
C.	Menon Identifies Fundamental Deficiencies In The Earlier Architecture	26
D.	The Proposed Combination Would Not Yield The Claimed System	27
E.	Petitioner's Theory Relies On Impermissible Hindsight	29
VI.	Menon Teaches Away from Petitioner's Proposed Combinations	31
A.	Express Criticism Of The First-Generation MultiMATLAB Architecture	32
B.	Incompatibility With Parallel MEX Extensions	33
C.	Teaching Away Undermines Any Motivation To Combine	34
VII.	Petitioner's Expert Testimony Cannot Cure Missing Claim Limitations	35
A.	Conclusory Assertions Are Not Substantial Evidence.....	36

IPR2025-00916 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,811,582

B.	Conflicts with the Written Record Must Be Resolved Against Petitioner	38
C.	Petitioner Is Bound by the Theories in the Petition	39
VIII.	The Cited References Are Not “Printed Publications”	40
A.	IBM Technical Manuals (RS/6000 and POEref, Exs. 1007–1008) 41	
B.	MPI Forum Draft Standard (“MPIref,” Ex. 1017)	42
C.	Cornell “MultiMATLAB” Web Article (“Trefethen,” Ex. 1006)	43
D.	SC’97 Conference Paper (“Menon,” Ex. 1005)	45
E.	Petitioner’s Citation Based Arguments Will Be Addressed Separately	47
IX.	Preserved Evidentiary Issues	47
X.	Conclusion	48

TABLE OF AUTHORITIES

	Page(s)
Cases	
<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)	43
<i>Blue Calypso, LLC v. Groupon, Inc.</i> , 815 F.3d 1331 (Fed. Cir. 2016)	40, 42, 44
<i>In re Cronyn</i> , 890 F.2d 1158 (Fed. Cir. 1989)	41, 42
<i>In re Cyclobenzaprine Hydrochloride Extended-Release Capsule Patent Litig.</i> , 676 F.3d 1063 (Fed. Cir. 2012)	29
<i>In re Dembiczak</i> , 175 F.3d 994 (Fed. Cir. 1999)	30
<i>DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.</i> , 567 F.3d 1314 (Fed. Cir. 2009)	27, 31, 32, 34
<i>In re Gurley</i> , 27 F.3d 551 (Fed. Cir. 1994)	31, 32, 34
<i>Intercontinental Great Brands LLC v. Kellogg N. Am. Co.</i> , 869 F.3d 1336 (Fed. Cir. 2017)	30
<i>In re Klopfenstein</i> , 380 F.3d 1345 (Fed. Cir. 2004)	42
<i>KSR Int’l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007).....	23, 24
<i>Kyocera Wireless Corp. v. Int’l Trade Comm’n</i> , 545 F.3d 1340 (Fed. Cir. 2008)	40

IPR2025-00916 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,811,582

<i>In re Lee</i> , 277 F.3d at 1343–45	38
<i>In re Lister</i> , 583 F.3d 1307 (Fed. Cir. 2009)	40
<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)	40, 42, 47
<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)	24
<i>Samsung Elecs. Co. v. Infobridge Pte. Ltd.</i> , 929 F.3d 1363 (Fed. Cir. 2019)	41, 43, 44
<i>TQ Delta, LLC v. Cisco Sys., Inc.</i> , 942 F.3d 1352 (Fed. Cir. 2019)	9, 15, 23, 36
Statutes	
35 U.S.C. § 316(e)	7, 8, 36
37 C.F.R. § 42.6(e).....	51
37 C.F.R. § 42.23(b)	39
37 C.F.R. § 42.24	50
37 C.F.R. § 42.24(d)	50
37 C.F.R. § 42.64(b)(1).....	7
37 C.F.R. § 42.65(a).....	37

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	<i>Allani v. Apple Inc.</i> , No. 6:24-cv-304-ADA (W.D. Tex. May 6, 2025) (D.I. 30)
2008	<i>Intellectual Ventures I LLC v. TCL Elects. Holdings Ltd.</i> , No. 6:23-cv-309-ADA (W.D. Tex. Sept. 10, 2024) (D.I. 44)
2009	<i>Acquis, LLC v. Hon Hai Precision Indus Co. Ltd.</i> , No. 6:23-cv-264-ADA (W.D. Tex. May 31, 2024) (D.I. 46)
2010	<i>ParkerVision, Inc. v. MediaTek Inc. et al.</i> , No. 6:22-cv-01163-ADA (W.D. Tex. April 29, 2024)
2011	<i>TOT Power Control, S.L. v. AT&T, Inc. et al.</i> , No. 6:21-cv-00107-ADA (W.D. Tex. June 6, 2025)

IPR2025-00916 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,811,582

<u>Exhibit</u>	<u>Description</u>
2012	<i>AlmondNet, Inc. v. Amazon.com, Inc., et al.</i> , 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	<i>Advanced Cluster Systems, Inc. v. Intel Corporation</i> , Order Appointing Technical Advisor (D.I. 71)
2018	<i>Advanced Cluster Systems, Inc. v. Advanced Micro Devices, Inc.</i> , Order Appointing Technical Advisor (D.I. 53)
2019	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)
2020	NVIDIA Company Overview (May 2025) (available at https://investor.nvidia.com/events-andpresentations/presentations/presentation-details/2025/Company-Overview-5-25/default.aspx)
2021	Intel Corporation’s Answer and Counterclaims to First Amended Complaint, <i>Advanced Cluster Systems, Inc. v. Intel Corporation</i> , 7:24-cv-00245-ADA (W.D. Tex. Feb. 5, 2025) (D.I. 41)
2022	Considered 02/18/2022 IDS Submitted in U.S. Patent Appln. No. 17/374,789

IPR2025-00916 PATENT OWNER RESPONSE
Inter Partes Review of U.S. Patent 11,811,582

<u>Exhibit</u>	<u>Description</u>
2023	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)
2024	<i>Advanced Cluster Systems, Inc. v. Intel Corporation</i> , Case No. 7:24-cv-00245-ADA (D.I. 79), Minute Entry Markman Hearing Proceedings Held 08/18/2025
2025	<i>VirtaMove, Corp. v. Microsoft Corp.</i> , Case No. 1:25-cv-00794-ADA (D.I. 33), Order Granting Motion to Dismiss
2026	<i>Yeti Coolers, LLC v. Waterbear Global, LLC</i> , Case No. 1:24-cv1251-ADA (D.I. 35, 35-1), Unopposed Motion to Extend Scheduling Order Deadlines and Proposed Order
2027	Intel Corporation's Third Amended Preliminary Invalidity And Subject-Matter Eligibility Contentions (Aug. 28, 2025)
2028 (New)	Patent Owner's Objections Under 37 C.F.R. § 42.64(b)(1) (Served Nov. 24, 2025)

I. INTRODUCTION

Patent Owner Advanced Cluster Systems, Inc. (“ACS”) hereby files this Response to the Petition in IPR2025-00916 challenging claims 1–20 of U.S. Patent No. 11,811,582 (“the ’582 Patent”). Petitioner contends that claims 1–3 and 5–19 would have been obvious under pre-AIA § 103 over Menon in view of Trefethen, RS6000, and MPIref, and that claims 4 and 20 would have been obvious over Menon in view of Trefethen, RS6000, MPIref, and POEref. The Petition’s obviousness grounds cannot withstand merits scrutiny. Institution was based on a preliminary record, with a final determination to be 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 ’582 Patent And The Claimed Distributed Computing Architecture

The ’582 Patent is directed to cluster computing systems that provide cluster computing functionality to high-level computing environments—particularly environments that use a kernel and support interactive workflows in which users issue commands and obtain results. (Ex. 1001, 1:16–17; 2:34–3:60.) The patent distinguishes cluster computing from distributed systems that merely run separate jobs on separate nodes without peer-level intercommunication by describing computer clusters as nodes that “intercommunicate so that the nodes can accomplish

a task as though they were a single computer,” and by emphasizing that many existing application programs are not designed to benefit from cluster intercommunication. (Ex. 1001, 1:22–29.) In the claimed systems, cluster node modules communicate tasks and data during computation, enabling coordinated multi-stage evaluation and result propagation across nodes, including by using MPI to “distribute tasks among the kernel modules” and provide communications “while the tasks are executing.” (Ex. 1001, 5:21–40; 6:19–25.)

The ’582 Patent’s disclosed architecture includes, *inter alia*, a user interface module, cluster node modules, and kernel modules. (Ex. 1001, 5:38–40, 11:19–27.) The written description explains that kernel modules include single-threaded program code and are associated with processors (Ex. 1001, 11:34–36), and that cluster node modules are initialized to provide cluster computing support and provide a way for kernel modules to communicate with one another (Ex. 1001, 11:41–44; 22:4–25). The claims recite this architecture by requiring, among other things, cluster node modules that communicate with kernels and with one another, and a multi-stage evaluation/result flow across processors—e.g., claim 1 requires a second kernel that executes matrix operations and “return[s] a first result,” a third kernel that “receive[s] the first result,” executes additional matrix operations on the first result, and “return[s] a second result,” and a first cluster node module configured to return a final result “based on the second result.” (Ex. 1001, 29:33–56.)

B. The Narrow, Dispositive Issues That Control This Case

This case turns on a focused set of dispositive issues that run through every instituted ground and every challenged claim. Patent Owner demonstrates that Petitioner's obviousness theories depend on rewriting the prior-art systems, importing undisclosed structures and execution semantics from MPI internals and later developments, and relying on conclusory expert assertions rather than what the cited references actually teach.

First, Petitioner cannot satisfy Claim 1's express result-return requirements, as addressed in Section IV.A. Claim 1 does not merely require distributed execution; it requires a system in which a second kernel executes matrix operations and returns a first result, a third kernel receives that first result and executes additional matrix operations on it and returns a second result, and a first cluster node module is configured to return a final result based on the second result. Menon and Trefethen do not disclose this behavior. As described in those references, `Eval()` forwards user commands for execution and may forward screen output to the interactive process, but they do not disclose any mechanism by which evaluation results are programmatically returned as computational objects in the manner required by Claim 1. Where data movement occurs in the prior art, it occurs only through separate, explicit operations such as `Put/Get` or `Send/Recv`, not through an implicit "Eval-return" mechanism. Petitioner's theory that `Eval()` returns results

sufficient to meet the claim relies entirely on expert inference that contradicts the implementation sections of the references and fills a gap the references themselves leave open. That failure alone defeats Claim 1 and every dependent claim that relies on returned results.

Second, Petitioner fails to establish the claimed multi-stage, dependent evaluation and result-propagation flow recited in Claim 1, as explained in Section IV.B. Claim 1 requires a specific ordered computation: the third kernel must receive the first result produced by the second kernel and perform additional matrix operations on that result, and the system must then return a final result based on that second result. Neither Menon nor Trefethen disclose a system in which one node's computed result is used as an input by another node as part of a defined, multi-stage evaluation flow, nor do they disclose a system-level return of a final result by the first cluster node module. Trefethen's cycle.m example describes a linear or ring-style progression that relies on console output or explicit messaging and does not disclose the dependent, multi-stage computation required by the claims. Petitioner's reliance on generic pipeline concepts and "could-have-arranged" narratives is impermissible hindsight reconstruction, not evidence that the prior art teaches or suggests the claimed evaluation flow.

Third, Petitioner fails to establish the claimed cluster configuration module of Claim 4, as addressed in Section IV.C. Claim 4 requires an internal cluster

configuration module that initializes cluster node modules as part of the claimed system architecture. Menon does not disclose any such module. Petitioner attempts to satisfy this limitation by pointing to IBM's Parallel Operating Environment (POE) and generic job-launch mechanisms, but POE is an external, system-level tool, not a module of the claimed cluster computing system. Job launching through external infrastructure is not the same as an internal cluster configuration module configured to initialize cluster node modules. Rebranding POE as part of the claimed architecture improperly collapses external environment functionality into system structure and cannot satisfy Claim 4.

Beyond these core structural deficiencies, Petitioner's grounds also fail for lack of a legally sufficient motivation to combine and reasonable expectation of success, as set forth in Section V. Petitioner's motivation theory for Menon and Trefethen reduces to the generalized assertion that a person of ordinary skill would have wanted to parallelize MATLAB more effectively and therefore would have combined those references. Such a high-level goal cannot substitute for a record-based rationale tied to the specific claimed configuration. Even as proposed, the combination still lacks multiple required claim elements, including result-return semantics, dependent multi-stage evaluation, and internal initialization modules.

Relatedly, as explained in Section VI, Menon teaches away from Petitioner's proposed combination. Menon criticizes the first-generation MultiMATLAB

architecture, identifies structural limitations, and presents a redesign rather than an extension. That discussion undermines both motivation to combine and any reasonable expectation of success and independently defeats obviousness.

As addressed in Section VII, Petitioner's expert testimony cannot cure these substantive defects. Dr. Bajaj repeatedly relies on inference, hindsight, and recharacterization of the references to supply missing claim limitations. Where expert narrative conflicts with the written record, the record controls. Conclusory *ipse dixit* is not substantial evidence and cannot carry Petitioner's burden in this proceeding.

In sum, when the prior art is read for what it actually discloses—not for what Petitioner and its expert now wish it disclosed—each of these narrow but critical issues is decided against Petitioner. Any one of them independently defeats the instituted grounds. Taken together, they demonstrate that the Petition's obviousness theories cannot withstand scrutiny on the full trial record, and that Claims 1–20 of the '582 Patent should be confirmed.

Separately, Petitioner has not proven public accessibility for “Menon” (Ex.1005), “Trefethen” (Ex.1006), “RS/6000” (Ex.1007), “POEref” (Ex.1008), and “MPIref” (Ex.1017). As such, they do not qualify as prior art, and all of Petitioner's invalidity grounds fail.

Finally, Patent Owner preserves its evidentiary and threshold objections to Petitioner's reliance on certain asserted references and supporting materials. Patent Owner has timely objected under 37 C.F.R. § 42.64(b)(1) to deficiencies in public accessibility, authentication, and foundation, including Petitioner's reliance on secondary evidence and expert narrative to establish facts not shown in the references themselves. These objections remain live for resolution on the full trial record and through any appropriate motion practice. Although the merits failures identified above independently defeat Petitioner's instituted grounds, the preserved evidentiary defects provide additional, independent bases on which the Board may decline to credit Petitioner's evidence or arguments.

C. Petitioner Cannot Meet Its Burden On Any Instituted Ground

Petitioner bears the burden to prove unpatentability by a preponderance of the evidence, and that burden never shifts. 35 U.S.C. § 316(e); *In re Magnum Oil Tools Int'l, Ltd.*, 829 F.3d 1364, 1375 (Fed. Cir. 2016). 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. The sections that follow address these failures directly, in the order that most efficiently resolves the instituted grounds.

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 20 at 2.) The Board instituted inter partes review of claims 1–20 of the ’582 Patent and instituted on all asserted grounds. (Paper 15, 43.) The instituted grounds allege that claims 1–3 and 5–19 would have been obvious over Menon in view of Trefethen, RS6000, and MPIref, and that claims 4 and 20 would have been obvious over Menon in view of Trefethen, RS6000, MPIref, and POEref. Patent Owner addresses the instituted grounds on the merits under the claims’ plain and ordinary meaning.

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 here, where Petitioner’s obviousness theories depend on attributing to Menon and Trefethen (i) programmatic return of evaluation results and (ii) a multi-stage, dependent evaluation flow across nodes—features that the references do not disclose.

D. Reliance on the Existing Evidentiary Record

Patent Owner does not submit an 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. 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). Patent Owner's Response is direct to claims in which the Board did not construe any terms when granting institution. (Paper 15, 9-10).

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

Petitioner advances two instituted obviousness grounds. The first ground alleges that claims 1–3 and 5–19 of the '582 Patent would have been obvious over Menon in view of Trefethen, RS6000, and MPIref. The second ground alleges that claims 4 and 20 would have been obvious over Menon in view of Trefethen, RS6000, and MPIref further in view of POEref. Although Petitioner relies on multiple references, it advances a unitary theory: that MultiMATLAB (Menon/Trefethen), implemented on IBM SP hardware (RS6000) and operating over MPI (MPIref), and—where applicable—supplemented with IBM's Parallel Operating Environment materials (POEref), discloses or renders obvious the '582 Patent's claimed cluster computing architecture, including multi-stage kernel execution with result propagation and return of a final result.

Those instituted theories fail for multiple independent reasons. The sections that follow address these structural defects in the order that resolves the instituted grounds most directly.

A. Petitioner Fails to Establish the Claimed Return of Evaluation Results Required by Claim 1

1. Claim 1 Requires Programmatic Return of Evaluation Results, Not Mere Execution or Display

Claim 1 of the '582 Patent is not satisfied by distributed execution alone. It requires a specific result-return architecture. In particular, claim 1 requires: (i) a second kernel that executes matrix operations and returns a first result using a second processor; (ii) a third kernel that receives the first result, executes additional matrix operations on that received result, and returns a second result using a third processor; and (iii) a first cluster node module configured to return a final result based on the second result. (Ex. 1001, 29:33-56.)

These return limitations are not incidental. They define how computational results propagate through the system and back to the originating component. The claims do not merely require that computations occur on multiple processors; they require that the system be structured so that results of those computations are programmatically returned as computational objects at defined stages of execution. A system that executes computations on remote nodes but does not return results in this manner does not satisfy the claim.

Accordingly, to meet its burden, Petitioner must identify prior-art disclosures showing that the asserted systems return evaluation results in the manner required by claim 1—not merely that computations are performed or that output is displayed.

2. Menon and Trefethen Do Not Disclose Programmatic Return of Evaluation Results

Petitioner's obviousness theory rests on the premise that MultiMATLAB's `Eval()` command returns the results of remote computations to the originating node or user interface. *See, e.g.*, Pet. 65 (stating "the final MATLAB process is programmed to output *the results* to the screen of the interactive process.") (emphasis added.) That premise is not supported by the cited references.

Trefethen describes `Eval()` only at a user-command level. The examples explain that when a user types an `Eval()` command, the command is executed on one or more MATLAB processes and any resulting *screen output* is forwarded to the interactive process as soon as it is ready. (Ex. 1006, 3–4.) Critically, Trefethen distinguishes between screen output and actual data movement. Trefethen explains that, at least in the implementation described, "we have not communicated between processes except to send screen output to the master process," and that when actual data transfer is required, it is accomplished through separate, explicit commands such as `Put`, `Get`, `Send`, and `Recv`. (Ex. 1006, 5–6.)

That disclosure directly undercuts Petitioner's theory. Forwarding screen output to the master console is not the same thing as returning a computational result as a data object that can be consumed by another kernel or used to generate a final result. Trefethen's own description makes clear that result values are not implicitly

returned through `Eval()`; they remain local to the executing process unless explicitly transmitted through separate messaging operations.

Menon adds no disclosure that cures this defect. Menon similarly describes `Eval()` as a mechanism for distributing commands to MATLAB processes over an underlying communication layer, but does not describe any mechanism by which the results of those commands are programmatically returned. (Ex. 1005, 3–6.) Like Trefethen, Menon treats explicit message-passing operations as the means by which data is transferred between processes. Nowhere does Menon disclose that the execution of an `Eval()` command itself causes a computed result to be returned to another kernel or to a cluster node module in the manner required by claim 1.

Thus, neither Menon nor Trefethen discloses a system in which execution of a remote computation returns a result as required by the claim.

3. Screen Output Is Not a “Returned Result” Under Claim 1

Petitioner repeatedly conflates user-visible screen output with the “return” of evaluation results required by Claim 1. That conflation is legally and technically incorrect.

Claim 1 requires that kernels *return* results that are subsequently *received* and used as inputs by other kernels, and that a cluster node module *return* a final result based on those returned values. Screen output is not a returned result in this sense. Screen output is a display artifact—it may show a value to a user, but it does not

constitute a programmatic return of a result to another system component for further computation.

The distinction is explicit in the prior art. Trefethen's description makes clear that screen output is forwarded automatically, while actual data transfer requires separate commands. (Ex. 1006, 5–6.) If `Eval()` returned computational results, there would be no need for `Put`, `Get`, `Send`, or `Recv`. The fact that those commands exist—and are described as the mechanisms for moving data—confirms that result return is not implicit.

Petitioner's attempt to treat console output as satisfying claim 1's return limitations rewrites the claim and contradicts the references. Claim 1 requires returned results that participate in subsequent computation, not values merely printed to a screen.

4. Petitioner's Eval-Return Theory Is Based on Expert *Ipse Dixit*, Not Disclosure

Because the references do not disclose programmatic return of evaluation results, Petitioner relies on expert testimony to supply the missing behavior. Dr. Bajaj asserts that a person of ordinary skill would understand MultiMATLAB to return results of computations executed via `Eval()`. But those assertions are unsupported by the cited references and directly contradict the implementation descriptions in Trefethen and Menon.

As the Federal Circuit has repeatedly held, conclusory expert testimony cannot supply missing claim elements. *TQ Delta*, 942 F.3d at 1359–60. Nor may “common sense” or expert intuition be used to fill a gap where the prior art is silent on the specific limitation at issue. *Arendi S.A.R.L.*, 832 F.3d at 1362–63.

Here, the gap is fundamental. The prior art does not disclose return of evaluation results in the manner required by claim 1. Petitioner’s expert cannot manufacture that disclosure by inference.

5. The Failure to Disclose Result Return Is Dispositive

Because Petitioner cannot establish that the asserted prior art discloses programmatic return of evaluation results, it cannot satisfy the express limitations of claim 1. That failure alone defeats the instituted grounds as to Claim 1 and all dependent claims that rely on returned results. Even before reaching the additional defects addressed in Sections IV.B–IV.C, Petitioner’s case fails at this threshold step.

B. Petitioner Fails to Establish the Claimed Multi-Stage, Dependent Evaluation Flow Required by Claim 1

1. Claim 1 Requires a Specific, Dependent Multi-Stage Evaluation Using Returned Results

Claim 1 of the ’582 Patent requires more than distributed computation across multiple processors. It recites a particular execution sequence with explicit computational dependencies. In particular, claim 1 requires: (i) a second kernel that

executes matrix operations and returns a first result using a second processor; (ii) a third kernel that *receives* the first result, executes additional matrix operations on that received result, *and* returns a second result using a third processor; and (iii) a first cluster node module configured to return a final result based on the second result. (Ex. 1001, 29:33-56.)

This language imposes a dependency constraint: the third kernel's computation must depend on the output of the second kernel, and the system must be structured to propagate that dependency through to a final result. A system in which multiple kernels execute independently, or in which results are merely displayed or manually retrieved, does not satisfy this limitation. To meet its burden, Petitioner must identify prior-art disclosures showing kernels configured to consume another kernel's computed result as an input to further computation, and a system-level mechanism that propagates that dependent result back to a final return.

2. Menon and Trefethen Do Not Disclose Kernels Using Other Kernels' Results as Inputs to Further Computation

Petitioner does not identify any disclosure in Menon or Trefethen in which a kernel receives a result computed by another kernel and performs additional matrix operations on that received result, as required by claim 1.

Trefethen's examples describe execution of commands on multiple MATLAB processes, often using `Eval()` to cause each process to execute a specified

command. (*See, e.g.*, Pet. 38 (citing Ex. 1006, 3–6 for “Eval(‘cycle’) script, Pet. 56-57 (cycle.m script), Pet. 65-67.) These examples show independent execution, not dependent execution. Each process executes the same or similar commands on its own data. Where data is transferred between processes, it is done explicitly through Send and Recv commands, typically as part of illustrative demonstrations of message passing—not as part of a defined, staged evaluation in which one process’s computed result is automatically used as input to another process’s computation. (Ex. 1006, 5–6.)

Menon likewise describes MultiMATLAB as a system in which commands are distributed to multiple MATLAB processes and executed there, but does not describe a computation in which the result of one kernel’s execution is automatically fed into another kernel’s execution as an input. (Ex. 1005, 3–6.) Menon’s discussion of MPI and MEX routines focuses on enabling parallel execution and communication, not on implementing a dependent, multi-stage evaluation pipeline in which kernels are configured to consume one another’s results.

In short, neither reference discloses kernels configured to perform the claim-required dependency: receiving another kernel’s computed result and executing additional matrix operations on that result.

3. Trefethen's `cycle.m` Example Does Not Disclose the Claimed Evaluation Chain

Petitioner relies heavily on Trefethen's `cycle.m` example to argue that the prior art discloses a chained, multi-node execution flow. *See, e.g.*, Pet. 57 (Ex. 1006, 6 (annotated); *id.* 65 (same)). That reliance is misplaced.

The `cycle.m` example demonstrates a simple sequence in which a value is passed along a ring of processes and modified at each step. (Ex. 1006, 6.) It does not disclose kernels configured to execute matrix operations on another kernel's returned result as part of a defined evaluation flow, nor does it disclose a system-level return of a final result by a first cluster node module. The example relies on explicit `Send` and `Recv` commands and is presented as an illustration of message passing, not as an architectural template for dependent evaluation in a cluster computing system.

More importantly, the example does not disclose the claim-required structure: it does not show a second kernel returning a first result, a third kernel receiving that result and executing additional matrix operations on it, and a first cluster node module returning a final result based on that second result. It is a demonstration of message passing, not a disclosure of the claimed multi-stage evaluation architecture.

Because Petitioner has not identified any disclosure of this required evaluation flow, whether through the use of `cycle.m` or elsewhere in the prior art, the proposed

combination cannot satisfy Claim 1. That failure independently defeats the instituted grounds, without regard to the additional defects addressed in Sections IV.C–IV.F.

C. Petitioner Fails to Establish the Claimed Cluster Configuration Module Required by Claim 4

1. Claim 4 Requires an Internal Cluster Configuration Module That Initializes Cluster Node Modules

Claim 4 of the '582 Patent adds a specific structural requirement to the system recited in Claim 1: it requires “a cluster configuration module that initializes at least one of the first cluster node module, the second cluster node module, or the third cluster node module.” (Ex. 1001, 30:1-4.) This limitation is not satisfied by any mechanism that merely launches processes or allocates hardware resources. It requires an internal module of the claimed system—i.e., a component of the cluster computing architecture itself—that performs initialization of cluster node modules so that they can operate together in the claimed environment.

The written description reinforces this distinction. The patent describes the cluster configuration module as part of the system’s architecture that communicates with cluster node modules to initialize them and enable cluster computing support. (Ex. 1001, 11:36–44, 22:4-25.) The claims thus draw a clear line between (i) external environment or operating-system functionality and (ii) internal system modules configured to initialize cluster node modules as part of the claimed architecture. To

meet its burden, Petitioner must identify prior art disclosing such an internal module—not merely an external job launcher or administrative tool.

2. Menon Does Not Disclose Any Cluster Configuration Module

Menon does not disclose a cluster configuration module as required by Claim 4. Menon describes MultiMATLAB as a system in which MATLAB processes are started on multiple processors and communicate using MPI, with one process designated as an interactive process and others waiting for commands. (Ex. 1005, 3–6.) Menon does not identify any internal software component responsible for initializing cluster node modules, nor does it describe any module that performs the functions attributed to the cluster configuration module in the '582 Patent.

To the contrary, Menon explains that in the implementation of MultiMATLAB designed specifically for the IBM SP2, “all MATLAB processes are started via POE, IBM’s Parallel Operating Environment.” (Ex. 1005, 6; Pet. 21 (noting that RS6000 is part of the IBM SP2 system).) That statement confirms that MultiMATLAB relies on an external system-level tool to launch processes. It does not disclose an internal module within MultiMATLAB that initializes cluster node modules as part of the application’s architecture. The absence of any such disclosure is dispositive: a claim requiring a specific module cannot be satisfied by a reference that does not disclose that module at all.

3. IBM's Parallel Operating Environment Is an External Job-Launch Tool, Not a Cluster Configuration Module

Petitioner attempts to fill the gap in Menon by relying on IBM's Parallel Operating Environment (POE) and associated documentation. (Pet. 113.) That reliance is misplaced.

POE is an external, system-level environment used to launch parallel jobs on IBM SP hardware. As described in the POE documentation, invoking the `poe` command causes the Partition Manager to allocate nodes, initialize the execution environment, and load the specified program on each node. (Ex. 1008, 46.) Those functions are performed by POE as part of the operating environment, not by any module of the application being launched.

Petitioner has not identified anything in the POE documentation to suggest that POE is part of MultiMATLAB's internal architecture, or that it functions as a module within a cluster computing system for purposes of initializing cluster node modules. POE operates outside the application; it does not configure or initialize application-level modules in the sense required by Claim 4. Treating POE as the claimed cluster configuration module collapses external job-launch infrastructure into internal system structure, contrary to the claim language and the patent's disclosure.

4. Job Launching Is Not the Same as Initializing Cluster Node Modules

Petitioner's theory rests on equating job launching with cluster configuration. That equation is incorrect. Launching processes on multiple nodes does not initialize cluster node modules as part of a cluster computing architecture; it merely causes programs to begin executing.

The '582 Patent distinguishes between these concepts. The cluster configuration module is a component of the system that initializes cluster node modules so that they can provide cluster computing support—i.e., it prepares the system's internal components to operate together in the claimed environment. (Ex. 1001, 11:36–44.) An external tool that allocates hardware resources and starts processes does not perform that role.

Petitioner's argument effectively reads the “cluster configuration module” limitation out of the claim by treating any mechanism that starts processes as satisfying the requirement. If that were correct, the limitation would be meaningless, because virtually every distributed application relies on some external mechanism to launch processes.

5. Petitioner's Theory Relies on Impermissible Gap-Filling and Expert *Ipse Dixit*

Because neither Menon nor POE discloses a cluster configuration module, Petitioner relies on expert testimony to assert that a person of ordinary skill would

have viewed POE as satisfying the claim. That assertion is unsupported by the references and constitutes impermissible gap-filling.

As explained above, conclusory expert testimony cannot supply missing claim elements. *TQ Delta*, 942 F.3d at 1359–60. Nor may an expert use “common sense” or intuition to fill a gap where the prior art is silent. *Arendi S.A.R.L.*, 832 F.3d at 1362–63. Here, the gap is clear: the asserted prior art does not disclose an internal cluster configuration module. Petitioner’s expert cannot manufacture that disclosure by recharacterizing an external job-launch environment as an internal system module.

6. The Failure to Disclose a Cluster Configuration Module Independently Defeats Claim 4

Because Petitioner has not identified any prior-art disclosure of a cluster configuration module that initializes cluster node modules as required by Claim 4, it cannot satisfy that limitation. This failure independently defeats Petitioner’s instituted grounds as to Claim 4, regardless of the additional defects addressed in Sections IV.A, IV.B.

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 Menon and Trefethen motivation-to-combine argument reduces to the unremarkable proposition that researchers wanted to parallelize high-level mathematical computing environments. Paper 16, 26-27 (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. 24 (asserting Menon "builds on" the MultiMATLAB system taught by Trefethen); Pet. 24 (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. 24 (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. 25-27 (asserting motivation to combine RS6000 with Menon because Menon uses the IBM SP2/RS6000 SP system); Pet. 111-112 (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, 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, 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.

Second, the asserted references' control and execution model is not the claimed execution model of the '582 Patent. Menon explicitly describes an architecture in which one process is "considered the interactive MATLAB" and "all other processes wait for commands from the interactive MATLAB process," i.e., a master-controlled dispatch model. (Ex. 1005, 6.) That disclosure does not describe the multi-stage, dependent evaluation and result-return architecture required by claim 1, which requires a second kernel to return a first result, a third kernel to receive that first result and return a second result, and a first cluster node module to return a final result based on the second result. (Ex. 1001, 29:33-56.) Petitioner cannot transform Menon's master-controlled architecture into the claimed system merely by pointing out that MPI supports point-to-point data transfers. The question is not whether MPI can transmit data between nodes in the abstract. The asserted references must disclose a system that performs the claimed ordered evaluation

sequence and returns the final result through the claimed components as recited. 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. 26-27 (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, 10 (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 as 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 LIMITATIONS

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 critical claim requirements, including attributing to MultiMATLAB an “Eval-return” mechanism and a three-node ordered evaluation sequence that are not disclosed by Trefethen or Menon. Those assertions are either unsupported by the cited references or directly contradicted by them. They therefore cannot carry Petitioner’s burden of proving unpatentability by a preponderance of the evidence.

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 in the context of Claim 1 of the ’582 Patent. 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 ¶¶262-263), even though Trefethen expressly distinguishes the forwarding of screen output from actual data transfer and identifies separate, explicit mechanisms—such as master-initiated put/get operations—for moving data between processes. (Ex. 1006, 5–6.) These are not reasonable inferences drawn from disclosed system behavior; they are conclusory statements offered to supply missing result-return and execution semantics that the asserted prior art does not disclose.

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 opinion that contradicts or fills gaps in what the reference actually teaches. *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. With respect to return of evaluation results, Petitioner’s “Eval-return” theory depends on treating console output or other user-visible output as the claim-required returned result, and on attributing to Eval () an implicit return-value mechanism. Yet Trefethen expressly distinguishes screen-output forwarding from actual data transfer, stating that (at least in the implementation described) there had been no inter-process communication “except to send screen output to the master process,” and it separately identifies explicit mechanisms—such as master-only put/get operations and Send/Recv—for moving data between processes. (Ex. 1006, 5–6.) Those disclosures are inconsistent with Petitioner’s suggestion that computational results are returned programmatically by default in a manner that satisfies claim 1’s

required “return” of results. Where the prior art 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 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 claim 1’s result-return and multi-stage dependent evaluation requirements relies on Menon’s interactive-process control model and Trefethen’s user-level scripts/output, supplemented by Dr. Bajaj’s characterizations. If that mapping fails on the merits—as it does—Petitioner cannot salvage the case by pivoting to a new interpretation of `Eval()` that supplies programmatic return of results, or by offering a new explanation of where the required dependent evaluation flow and final-result return

are disclosed. Those would be new theories and would improperly expand 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 ground 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 limitations, cannot contradict the written record, and cannot substitute for the disclosures that Petitioner was required to prove.

VIII. THE CITED 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) qualifies 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 ¶¶132, 136-137.) 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 ¶¶142-144. 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. (Pet.; Ex. 1003 ¶¶129-130.) 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.

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. 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 ¶129; 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 2001 and was cited by numerous subsequent papers. *Id.*; Ex. 1003 ¶¶123-125; 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. 14–15; Ex. 1003 ¶¶123-124.) 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 ¶123.) 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. 2028 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 claim-required programmatic return of evaluation results or the multi-stage, dependent evaluation flow recited in Claim 1—i.e., a second kernel returning a first result, a third kernel receiving that first result and returning a second result, and a first cluster node module returning a final result based on the second result.

Petitioner's attempt to supply these missing limitations through expert narrative is legally insufficient, as conclusory expert assertions cannot substitute for what the references fail to disclose. Moreover, Menon's own critique of the first-generation MultiMATLAB architecture undermines Petitioner's combination theory and confirms the absence of a record-based rationale and reasonable expectation of success for achieving the claimed system. Additionally, the asserted prior art fails to disclose all of the elements in Claim 4. Accordingly, the Board should determine that Petitioner has not proven unpatentability, and the challenged claims should be confirmed.

Nor has Petitioner carried its distinct burden to show a record-based motivation to combine these references with a reasonable expectation of success; Menon's own critique of the first-generation MultiMATLAB architecture teaches

away from using that system as the very foundation for the claimed architecture, and confirms that the straightforward path was to abandon, not extend, the earlier design.

Separately, and as a merits issue, Petitioner has failed to carry its burden to establish that certain asserted references qualify as printed publications. Petitioner has not shown the requisite public accessibility at the relevant time. In addition, Patent Owner has preserved evidentiary objections to deficiencies in authentication, foundation, and the use of unsupported secondary evidence. Even if the Board were to reach these issues independently, they provide additional grounds for declining to credit Petitioner's theories.

On this evidentiary record, the Board should confirm claims 1–20 of the '582 Patent.

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 10,490 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

Dated: February 6, 2026

Respectfully submitted,

/David P. Lindner/
David P. Lindner, Reg. No. 53,222
Counsel for Patent Owner
Advanced Cluster Systems, Inc.