

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

**PETITIONER'S REPLY
TO PATENT OWNER'S RESPONSE**

TABLE OF CONTENTS

PETITIONER’S EXHIBIT LIST	iv
I. INTRODUCTION.....	1
II. CLAIM CONSTRUCTION.....	1
A. The specification provides a clear definition of “hardware processor with a plurality of processing cores”	1
B. Patent Owner excludes hardware processor configurations described in the specification	7
C. The “plurality of processing cores” language does not limit the “hardware processor” to only a “multi-core microprocessor”	9
D. The Board should reject Patent Owner’s arguments	10
III. THE PETITION ESTABLISHES THAT THE CHALLENGED CLAIMS ARE OBVIOUS	11
A. The “Symmetric MultiProcessor (SMP)” in RS6000 teaches a “ <i>hardware processor with a plurality of processing cores</i> ”	11
1. RS6000 teaches a processor having multiple microprocessors each having at least one core.....	11
2. Even applying Patent Owner’s arguments, RS6000 teaches a single chip package (Power3) having 8 cores.....	12
B. Menon/Trefethen teaches the “1→2→3→1” order of operations in the Challenged Claims.....	13
1. Menon/Trefethen’s MultiMATLAB system performs the claimed order of operations.....	13
2. Patent Owner’s attempts to distinguish Menon/Trefethen fail.....	17
C. The Petition establishes that combining Menon and Trefethen would have been obvious	24
1. Menon and Trefethen teach how to implement MultiMATLAB	24
2. Menon does not teach away from the combination.....	25

IV. THE REFERENCES CITED IN THE PETITION ARE PRINTED PUBLICATIONS 27

1. Menon is a printed publication..... 27

2. Trefethen is a printed publication..... 29

3. RS6000 is a printed publication 31

4. POEref is a printed publication 32

5. MPIref is a printed publication..... 33

V. THE PRIOR ART REFERENCES CITED IN THE PETITION ARE ADMISSIBLE 35

VI. CONCLUSION 35

CERTIFICATE OF SERVICE 37

PETITIONER'S EXHIBIT LIST

Ex.1001	U.S. Patent No. 11,570,034
Ex.1002	Prosecution History of U.S. Patent No. 11,570,034
Ex.1003	Declaration of Dr. Chandrajit L. Bajaj under 37 C.F.R. § 1.68
Ex.1004	<i>Curriculum Vitae</i> of Dr. Chandrajit L. Bajaj
Ex.1005	"MultiMATLAB: Integrating MATLAB with High-Performance Parallel Computing," Menon et al., SC '97: Proceedings of the 1997 ACM/IEEE Conference on Supercomputing 1997
Ex.1006	"MultiMATLAB: MATLAB on Multiple Processors," Trefethen et al., 1996 Computer Science Technical Report, Cornell University 1996
Ex.1007	"RS/6000 SP: Planning Vol. 1, Hardware and Physical Environment," IBM 2001
Ex.1008	"Operation and Use, Volume 1, Using the Parallel Operating Environment," IBM 2001
Ex.1009	"Single program multiple data" in Dictionary of Algorithms and Data Structures, P. E. Black, (accessible at https://www.nist.gov/dads/HTML/singleprogrm.html) December 2004
Ex.1010	The RS/6000 SP Inside Out" to Barrios et al., IBM 1999
Ex.1011	"High Performance Cluster Computing: Programming and Applications," Rajkumar Buyya, Vol. 2, 1999
Ex.1012	"Analysis of 100Mb/s Ethernet for the Whitney Commodity Computing Testbed," Fineberg et al., NAS Technical Report NAS-97-025 1997
Ex.1013	"A Parallel Linear Algebra Server for Matlab-like Environments," G. Morrow et al., SC '98: Proceedings of the 1998 ACM/IEEE Conference on Supercomputing 1998
Ex.1014	"Parallel MATLAB: Doing It Right," R. Choy et al., Proceedings of the IEEE, Vol. 93, No. 2, 2005
Ex.1015	"Mathematica Parallel Computing Toolkit - Unleash the Power of Parallel Computing," R. Maeder, 2005

Ex.1016	“Mastering MATLAB® 5 – A Comprehensive Tutorial and Reference,” D. Hanselman et al., 1998
Ex.1017	“MPI: A Message-Passing Interface Standard,” Message Passing Interface Forum, 1994
Ex.1018	“Modern Operating Systems,” A.S. Tanenbaum, 2001
Ex.1019	September 26, 2024 Complaint in Western District of Texas against Intel
Ex.1020	Order Granting Joint Motion for Entry of Amended Scheduling Order for <i>Advanced Cluster Systems, Inc. v. Intel Corporation</i> , 7:24-CV-00245 (W.D.TX.)
Ex.1021	Federal Court Statistics
Ex.1022	Advanced Cluster Systems, Inc.’s Preliminary Infringement Contentions, January 22, 2005, <i>Advanced Cluster Systems, Inc. v. Intel Corporation</i> , 7:24-cv-00245 (WDTX)
Ex.1023	Cornell Websites
Ex.1024	“pyMPI–An introduction to parallel Python using MPI,” P. Miller 2002
Ex.1025	Amended Joint Claim Construction Chart for <i>Advanced Cluster Systems, Inc. v. NVIDIA Corporation</i> , 1:19-cv-02032 (DDE)
Ex.1026	SC97: High Performance Networking and Computing Conference Website
Ex.1027	Declaration of IEEE
Ex.1028	ACM Digital Library - Citations to Menon
Ex.1029	Affidavit of Archive.org
Ex.1030	HPC Wire Article about the SC97 High Performance Computing Conference
Ex.1031	Flyer about the SC97 High Performance Computing Conference
Ex.1032	IEEE Xplore Website
Ex.1033	Article about Cornell Theory Center
Ex.1034	ACM Digital Library
Ex.1035	ACM Digital Library - Citations to Menon

Ex.1036	A Beginner's Guide to the IBM SP
Ex.1037	ACM Digital Library - Citations to MPIref
Ex.1038	IBM Website for RS6000
Ex.1039	IBM Website for POE
Ex.1040	ACS Preliminary Response, IPR2021-00019
Ex.1041	Decision Denying Institution, IPR2021-00019
Ex.1042	Sotera Stipulation, <i>Advanced Cluster Systems, Inc. v. Intel Corporation</i> , 7:24-cv-00245 (WDTX)
Ex.1043	September 26, 2024 Complaint against AMD in Western District of Texas
Ex.1044	Scheduling Order for <i>Advanced Cluster Systems, Inc. v. Advanced Micro Devices, Inc.</i> , 7:24-CV-00244 (W.D.TX.)
Ex.1045	Advanced Cluster Systems, Inc.'s Preliminary Infringement Contentions, February 24, 2025, <i>Advanced Cluster Systems, Inc. v. Advanced Micro Devices, Inc.</i> , 7:24-cv-00244 (WDTX)
Ex.1046	Sotera Stipulation, <i>Advanced Cluster Systems, Inc. v. Advanced Micro Devices, Inc.</i> , 7:24-cv-00244 (WDTX)
Ex.1047	Declaration of International Business Machines Corp.
Ex.1048	Affidavit of Tanya Zeif, Internet Archive
Ex.1049	MPIref at Argonne National Laboratory
Ex.1050	"Nvidia, AMD CEOs Rally Behind President Trump's AI 'Action Plan'," B. Lin, Wall Street Journal, July 24, 2025.
Ex.1051	"Microsoft offers cloud customers AMD alternative to Nvidia AI processors," M. Cherney, Reuters, May 17, 2024.
Ex.1052	"What milestone advancements in computer chips mean for the military," Breaking Defense
Ex.1053	"Lawrence Livermore National Laboratory's El Capitan verified as world's fastest supercomputer," LLNL
Ex.1054	"Economic Contributions of Data Centers in the United States, 2017-2023," PWC, Feb. 2025

Ex.1055	Senate hearing on “Winning the AI Race: Strengthening U.S. Capabilities in Computing and Innovation,” May 8, 2025
Ex.1056	“DoD Extends Contract for ORAN-based Private 5G Networks through 2025,” Echostar, March 18, 2024
Ex.1057	“HPE selected to deliver supercomputers for DoD computing modernization program,” M. Iriarte, Military Embedded Systems, February 20, 2018
Ex.1058	“ZMicro to Provide Rugged Computers for US Sky Warden Aircraft,” R. Manuel, The Defense Post, Oct. 6, 2022
Ex.1059	“Argonne National Laboratory’s Aurora 2+ ExaFLOPS System Will Enable New Science and Engineering,” Intel
Ex.1060	“Intel battles AMD with new data center chips,” Cherney et al, Reuters, June 4, 2024
Ex.1061	“X86 vs. ARM: A Deep Dive into the Architecture,” Semicon Electronics, October 28, 2024
Ex.1062	U.S. Patent No. 8,601,101
Ex.1063	Amended Order Assigning the Business of the Court (Judge Moses, Jan. 31, 2025)
Ex.1064	LexMachina docket report
Ex.1065	LexMachina docket report
Ex.1066	LexMachina docket report
Ex.1067	LexMachina docket report
Ex.1068	“TensorWave just deployed the largest AMD GPU training cluster in North America — features 8,192 MI325X AI accelerators tamed by direct liquid-cooling,” Tom’s Hardware, July 14, 2025
Ex.1069	“Intel and Trump Administration Reach Historic Agreement to Accelerate American Technology and Manufacturing Leadership,” Aug. 22, 2025
Ex.1070	LexMachina docket report

Ex.1071	March 24, 2025 Memorandum titled “Guidance on USPTO’s recission of ‘Interim Procedure for Discretionary Denials in AIA Post-Grant Proceedings with Parallel District Court Litigation’” (Chief Administrative Patent Judge Boalick)
Ex.1072	“FAQs for Interim Processes for PTAB Workload Management” (USPTO)
Ex.1073	March 26, 2025 Memorandum titled “Interim Processes for PTAB Workload Management” (Acting Director Stewart)
Ex.1074	Aug. 13, 2025 Memo “Interim Director Discretionary Process” (USPTO)
Ex.1075	“Oracle and AMD Collaborate to Help Customers Deliver Breakthrough Performance for Large-Scale AI and Agentic Workloads,” Oracle, June 12, 2025.
Ex.1076	“How AMD is Intensifying AI Chip Focus Amid Global Tech Race,” Technology Magazine, Nov. 17, 2024.
Ex.1077	“AMD’s AI Chips Gain Ground in Data Centers: A Sign for More Upside?,” Zacks Equity Research, July 16, 2025.
Ex.1078	“Transcript: Sam Altman Testifies At US Senate Hearing On AI Competitiveness,” TechPolicy.Press, May 8, 2025.
Ex.1079	“Meta's Obsession With AI Is Great News For AMD,” Seeking Alpha, July 9, 2025.
Ex.1080	U.S. Patent Publication No. 2005/0021751
Ex.1081	U.S. Patent No. 6,074,427
Ex.1082	“USPTO rescinds memorandum addressing discretionary denial procedures,” USPTO, Feb. 28, 2025
Ex.1083	Reserved
Ex.1084	Reserved
Ex.1085	Claim Construction Order; <i>Advanced Cluster Systems, Inc. v. Advanced Micro Devices, Inc., Intel Corporation</i> ; Case No. 7:24-cv-00244-ADA and 7:24-cv-00245-ADA; Dkt.87; Filed 09/26/25
Ex.1086 (New)	Declaration Of Ingrid Hsieh-Yee, Ph.D.

Ex.1087 (New)	CV Of Ingrid Hsieh-Yee, Ph.D.
Ex.1088 (New)	Nilges, <i>The Online Computer Library Center's Open WorldCat Program</i>, Library Trends (2006)
Ex.1089 (New)	ACM copy of Menon
Ex.1090 (New)	WorldCat.org record for SC'97
Ex.1091 (New)	SC'97 Academia Sinica bibliography
Ex.1092 (New)	Internet Archive copy of Menon (archived 1998)
Ex.1093 (New)	Archived webpages related to SC'97
Ex.1094 (New)	Harrop et al., <i>Supporting runtime tool interaction for parallel simulations</i>, SC'98: Proceedings of the 1998 ACM/IEEE Conference on Supercomputing (pp. 37-37) ("Harrop")
Ex.1095 (New)	Rana et al., <i>An XML based component model for generating scientific applications and performing large scale simulations in a meta-computing environment</i>, International Symposium on Generative and Component-Based Software Engineering (pp. 210-224) ("Rana")
Ex.1096 (New)	Almasi et al., <i>MaJIC: A MATLAB just-in-time compiler</i>, International Workshop on Languages and Compilers for Parallel Computing (pp. 68-81) ("Almasi")
Ex.1097 (New)	Houstis et al., <i>On the Future of Problem Solving Environments</i>, Department of Computer Science Technical Reports, Report number 00-009, Paper 1487 ("Houstis")
Ex.1098 (New)	Internet Archive copy of Trefethen (archived May 1996)
Ex.1099 (New)	Cornell copy of Trefethen

Ex.1100 (New)	Trefethen Internet Archive materials
Ex.1101 (New)	Trefethen bibliography Washington State University Library
Ex.1102 (New)	Lang, <i>A multi-threaded extension to a high level interactive statistical computing environment</i>, Doctoral Dissertation, University of California, Berkeley (Fall 1997)
Ex.1103 (New)	Lang dissertation bibliography University of California, Berkeley Library
Ex.1104 (New)	Quinn et al., <i>Preliminary results from a parallel MATLAB compiler</i>, Proceedings of the First Merged International Parallel Processing Symposium and Symposium on Parallel and Distributed Processing (pp. 81-87) (1998)
Ex.1105 (New)	Quinn bibliography Linda Hall Library
Ex.1106 (New)	Quinn bibliography Michigan Technological University
Ex.1107 (New)	Chauveau et al., <i>Menhir: An environment for high performance Matlab</i>, International Workshop on Languages, Compilers, and Run-Time Systems for Scalable Computers (pp. 27-40) ("Chauveau")
Ex.1108 (New)	Chauveau Bibliography Library of Congress
Ex.1109 (New)	Internet Archive copy of RS6000 (June 17, 2001)
Ex.1110 (New)	Archived webpages showing path to RS6000
Ex.1111 (New)	Archived webpages showing alternate path to RS6000
Ex.1112 (New)	Moncrief, <i>IBM RS/6000</i>, available at https://users.cs.jmu.edu/abzugcx/Public/Student-Produced-Term-Projects/Computer-ARCHITECTURE-2002-SUMMER/IBM-RS-6000-by-Robert-Moncrief-2002-SUMMER-CS-585.doc

Ex.1113 (New)	Archived webpages showing path to POEref
Ex.1114 (New)	Saif et al., <i>Understanding the behavior and performance of non-blocking communications in MPI</i>, European Conference on Parallel Processing (pp. 173-182) (August 2004) (“Saif”)
Ex.1115 (New)	Saif bibliography Library of Congress
Ex.1116 (New)	U.S. Pub. No. 2005/0078605 to Huang et al. (“Huang”)
Ex.1117 (New)	Internet Archive copy of MPIref (July 3, 1998)
Ex.1118 (New)	Archived webpages showing path to MPIref
Ex.1119 (New)	Gropp et al., <i>Implementing MPI: the 1994 MPI implementors' workshop</i>, Proceedings Scalable Parallel Libraries Conference (pp. 55-59) (October 1994)
Ex.1120 (New)	MPIref bibliography National Science Library of Canada
Ex.1121 (New)	MPIref bibliography University of Edinburgh
Ex.1122 (New)	Majumdar, <i>Fibers: A high performance MPMD mechanism for integrated asynchronous data and control transfer</i>, Doctoral Dissertation, Yale University (1994)
Ex.1123 (New)	Majumdar bibliography Yale University Library
Ex.1124 (New)	Huber Jr. et al., <i>PPFS: A high performance portable parallel file system</i>, Proceedings of the 9th International Conference on Supercomputing (pp. 385-394) (1995)
Ex.1125 (New)	Huber Jr. bibliography ACM Digital Library

Ex.1126 (New)	Bhat at al., <i>Issues in using heterogeneous HPC systems for embedded real time signal processing applications</i>, Proceedings of the 2nd International Workshop on Real-Time Computing Systems and Applications (RTCSA '95) (October 1995)
Ex.1127 (New)	Bhat bibliography IEEE
Ex.1128 (New)	Worley, <i>MPI performance evaluation and characterization using a compact application benchmark code</i>, Proceedings Second MPI Developer's Conference (pp. 170-177) (1996)
Ex.1129 (New)	Worley bibliography Linda Hall Library
Ex.1130 (New)	Karpievitch et al., <i>mGrid: A load-balanced distributed computing environment for the remote execution of the user-defined Matlab code</i>, BMC Bioinformatics 2006, 7:139 (March 15, 2006)
Ex.1131 (New)	Pfarrhofer et al., <i>Distributed Optimization of Fiber Optic Network Layout Using MATLAB</i>, ICCSA 2004, LNCS 3045, pp. 538-547 (2004)
Ex.1132 (New)	Elphick et al., <i>Partial Evaluation of MATLAB</i>, GPCE 2003, LNCS 2830, pp. 344-363 (2003)
Ex.1133 (New)	Fernández et al., <i>Performance of Message-Passing MATLAB Toolboxes</i>, VECPAR 2002, LNCS 2565, pp. 228-242 (2003)
Ex.1134 (New)	Bekas et al., <i>Parallel Computation of Pseudospectra using Transfer Functions on a MATLAB-MPI Cluster Platform</i>
Ex.1135 (New)	Bekas et al., <i>Parallel Computation of Pseudospectra Using Transfer Functions on a MATLAB-MPI Cluster Platform</i>, Euro PVM/MPI 2002, LNCS 2474, pp. 199-207 (2002)
Ex.1136 (New)	Rana et al., <i>Agent based Service Integration for Distributed Problem Solving Environments</i>, Proceedings of the 34th Hawaii International Conference on System Sciences (2001)
Ex.1137 (New)	Li et al., <i>Wrapping MPI-based legacy codes as Java/CORBA components</i>, Future Generation Computer Systems 18, pp. 213-223 (2001)
Ex.1138 (New)	Almasi et al., <i>MaJIC: A Matlab Just-In-time Compiler</i>, LCPC2000, LNCS 2017, pp. 68-81 (2001) (duplicate of Ex.1096)

Ex.1139 (New)	Rana et al., <i>Implementing Problem Solving Environments for Computational Science</i>, Euro-Par 2000, LNCS 1900, pp. 1345-1349 (2000)
Ex.1140 (New)	Rana et al., <i>An XML Based Component Model for Generating Scientific Applications and Performing Large Scale Simulations in a Meta-computing Environment</i>, GCSE'99, LNCS 1799, pp. 210-224 (2000) (duplicate of Ex.1095)
Ex.1141 (New)	Menon et al., <i>A Case for Source-Level Transformations in MATLAB</i>, Proceedings of DSL'99: The 2nd Conference on Domain-Specific Languages, Austin, Texas, USA (October 3-6, 1999)
Ex.1142 (New)	Menon (ACM copy)
Ex.1143 (New)	Casanova et al., <i>Applying Netsolve's Network-Enabled Server</i>, IEEE Computation, Science & Engineering, pp. 57-67 (July-September 1998)
Ex.1144 (New)	Bikshandi et al., <i>Programming for Parallelism and Locality with Hierarchically Tiled Arrays</i>, PPOPP'06 New York, pp. 48-57 (March 29-31, 2006)
Ex.1145 (New)	Guo et al., <i>Hierarchically Tiled Arrays for Parallelism and Locality</i>, IEEE (2006)
Ex.1146 (New)	Kepner et al., <i>MatlabMPI</i> (Jan 3, 2003)
Ex.1147 (New)	Fernández et al., <i>Performance of Message-Passing MATLAB Toolboxes</i>, VECPAR 2002, LNCS 2565, pp. 228-242 (2003) (duplicate of Ex.1133)
Ex.1148 (New)	Banerjee et al., <i>A MATLAB Compiler For Distributed, Heterogeneous, Reconfigurable Computing Systems</i>, IEEE (2000)
Ex.1149 (New)	Haldar et al., <i>Match Virtual Machine : An Adaptive Runtime System to execute MATLAB in Parallel</i>, IEEE (2000)
Ex.1150 (New)	Chauveau et al., <i>Menhir: An environment for high performance Matlab</i>, Scientific Programming 7, pp. 303-312 (1999)

Ex.1151 (New)	Wu et al., <i>Distributed Data Structure Design for Scientific Computation</i>, ICS 98 Melbourne Australia, pp. 227-234 (1998)
Ex.1152 (New)	Chen et al., <i>Fault Tolerant High Performance Computing by a Coding Approach</i>, PPOPP'05 Chicago (June 15-17, 2005)
Ex.1153 (New)	Bader et al., <i>Practical Parallel Algorithms for Dynamic Data Redistribution, Median Finding, and Selection</i>, Proceedings of IPPS '96, pp. 292-301 (1996)
Ex.1154 (New)	Bustamante et al., <i>Efficient Wire Formats for High Performance Computing</i>, IEEE (2000)
Ex.1155 (New)	Sirotkin et al., <i>Parallel Schwarz methods for convection-dominated semilinear diffusion problems</i>, Journal of Computational and Applied Mathematics 145, pp. 189-211 (2002)
Ex.1156 (New)	Hascoët et al., <i>Automatic Placement of Communications in Mesh-Partitioning Parallelization</i>, PPOPP '97 Las Vegas, pp. 136-144 (1997)
Ex.1157 (New)	Park et al., <i>Efficient Algorithms for Block-Cyclic Array Redistribution between Processor Sets</i>, SC '98 Orlando, pp. 1-13 (1998)
Ex.1158 (New)	Chang et al., <i>Low-Latency Communication on the IBM RISC System/6000 SP</i>, IEEE (1996)
Ex.1159 (New)	Karamceti et al., <i>Software Overhead in Messaging Layers: Where Does the Time Go?</i>, ASPLOS VI - 10/94 San Jose, pp. 51-60 (1994)
Ex.1160 (New)	Wong et al., <i>Architectural Requirements and Scalability of the NAS Parallel Benchmarks</i>, SC '99 Portland (1999)
Ex.1161 (New)	Xu et al., <i>Optimal Software Multicast in Wormhole-Routed Multistage Networks</i>, IEEE (1994)
Ex.1162	AMD Settlement Agreement (filed under seal)
Ex.1163 (New)	Supplemental Declaration of Dr. Chandrajit L. Bajaj under 37 C.F.R. § 1.68
Ex.1164 to Ex.1167	Reserved

Ex.1168 (New)	MIPS32™ Architecture For Programmers – Volume I: Introduction to the MIPS32™ Architecture (March 12, 2001)
--------------------------	---

I. INTRODUCTION

The Petition, supported by expert testimony (Ex.1003), establishes how the prior art renders obvious claims 1-30 of the '034 patent (the "Challenged Claims"). The Patent Owner Response ("POR") does nothing to rebut the teachings of the prior art because its arguments (1) rely on a construction of "hardware processor" that is inconsistent with the specification; (2) imports limitations into the claims; and (3) mischaracterizes the prior art's teachings.

II. CLAIM CONSTRUCTION

A. The specification provides a clear definition of "hardware processor with a plurality of processing cores"

The Board's Institution Decision ("DI," Paper 16) ordered the parties to "propose and support a particularized plain meaning or other construction" of the "hardware processor" limitations of independent claims 1 and 30,¹ "with particular attention to the description of a hardware processor with a plurality of processing cores in the '034 patent." DI, 14 (citing Ex.1001, 5:11-37, 7:14-36, 9:50-10:3).

Petitioner submits that the plain and ordinary meaning of "*a hardware processor with a plurality of processing cores*" is "**one or more single-core or**

¹ The proposed construction would also apply to independent claims 25 and 28, which recite a "hardware processor with a plurality of processing cores."

multi-core microprocessors with a plurality of processing cores.”² See Ex.1001, 5:11-37, 7:14-36, 9:50-10:3; Ex.1163, ¶9.

This construction is adapted directly from the specification, which creates a specific relationship between “processor” and “processor cores,”³ while drawing a clear distinction between “processor” and “microprocessor.” Ex.1163, ¶10.

First, column 5 provides language nearly identical to the claim language at issue:

nications architecture such as a local bus (not shown). A computer system can include one or more processors, and 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 15 one can be included in a single chip package. Software code

Ex.1001, 5:11-16.

² *Italics* denote claim language. **Bold** text in quotes denotes emphasis. All emphasis added unless otherwise indicated.

³ The specification uses the terms “processoror cores” and “processing cores” interchangeably. See, e.g., Ex.1001, 5:11-16, 5:25-30 (referring to “processor cores”), 7:27-29, 8:46-48 (referring to “processing cores”); Ex.1163, ¶11.

Here, the specification describes a “processor” as “contain[ing] one or more processor cores.” In turn, those “processor cores” are described as “generally independent microprocessors”:

communications architecture such as a local bus (not shown). A computer system can include one or more processors, and 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 15 one can be included in a single chip package. Software code

Ex.1001, 5:11-16; Ex.1163, ¶13.

Thus, the specification creates the relationship that a “processor” can “**contain** one or more” “independent microprocessors.” Ex.1163, ¶14.

Fig. 1 depicts three hardware processors (e.g., “processor 112a,” “processor 112b,” and “processor 132”):

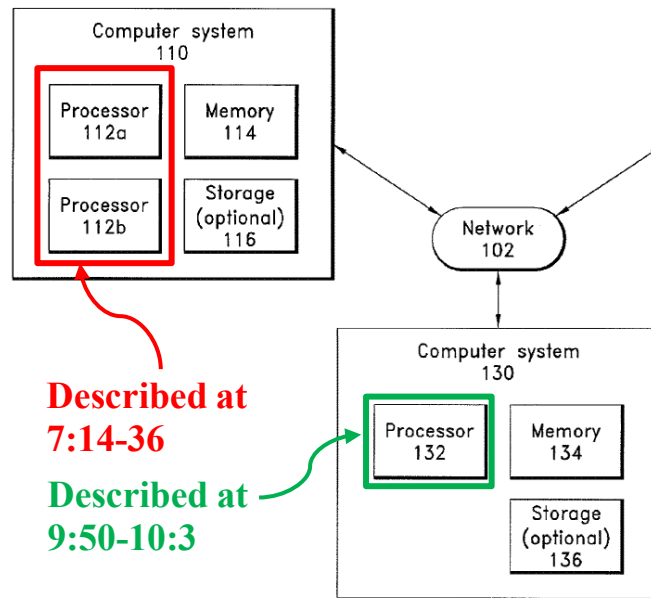


FIG. 1

Ex.1001, Detail of FIG. 1 (annotated); Ex.1163, 15.

Processors 112a-b of Fig. 1 are described in column 7:

1. Processors **112a-b**

In one embodiment, the computer system **110** includes one or more processors **112a-b**. The processors 112a-b can be one or more general purpose single-core or multi-core microprocessors such as, for example, a Pentium® processor, a Pentium® II processor, a Pentium® Pro processor, a Pentium® III processor, Pentium® 4 processor, a Core Duo® processor, a Core 2 Duo® processor, a Xeon® processor, an Itanium® processor, a Pentium® M processor, an x86 processor, an Athlon® processor, an 8051 processor, a MIPS® processor, a PowerPC® processor, an ALPHA® processor, etc. In addition, one or more of the processors

Ex.1001, 7:14-25 (annotations added). Notably, just as described in column 5, the

specification states that a “processor” can be “one or more general purpose single-core or multi-core microprocessors.”⁴ Ex.1001, 7:14-36; Ex.1163, ¶16.

Processor 132 is discussed in column 9, which describes the same distinction and relationship between “processor” and “microprocessors”:

In one embodiment, the computer system **130** includes a processor **132**. The processor 132 can be a general purpose single-core or multi-core microprocessors such as a Pentium® processor, a Pentium® II processor, a Pentium® Pro processor, a Pentium® III processor, Pentium® 4 processor, 55 a Core Duo® processor, a Core 2 Duo® processor, a Xeon® processor, an Itanium® processor, a Pentium® M processor, an x86 processor, an Athlon® processor, an 8051 processor, a MIPS® processor, a PowerPC® processor, or an ALPHA® processor. In addition, the processor **132** can be 60

Ex.1001, 9:51-60.

As with column 5 and column 7, the specification describes that a “processor” can be “microprocessors” (plural). Ex.1001, 9:50-10:3; Ex.1163, ¶17-18.

⁴ This language refers to each of “processors 112a-b” individually rather than collectively, because allowing the plural “processors 112a-b” to collectively be “one...microprocessor” is non-sensical and would conflict with column 5’s description. Ex.1001, 7:16-18; Ex.1163, ¶16.

Because the specification describes that (i) a processor can contain one or more independent microprocessors (Ex.1001, 5:11-16), (ii) a processor can be one or more single-core or multi-core microprocessors (Ex.1001, 7:14-36), and (iii) a processor can be multiple microprocessors (Ex.1001, 9:50-10:3), the plain and ordinary meaning of “*a hardware processor with a plurality of processing cores*” is “**one or more single-core or multi-core microprocessors with a plurality of processing cores.**” Ex.1163, ¶19. This construction is adapted directly from column 7, which states that each of the “processors” (each a “hardware processor”) “can be one or more general purpose single-core or multi-core microprocessors.” Ex.1001, 7:16-18; Ex.1163, ¶20.⁵

Accordingly, the Board should construe the claim language “*a hardware processor with a plurality of processing cores*” to mean “**one or more single-core or multi-core microprocessors with a plurality of processing cores**” consistent with the explicit disclosure of the specification. *See* Ex.1001, 5:11-37, 7:14-36, 9:50-10:3; Ex.1163, ¶¶21-23.

⁵ The proposed construction removes the term “general purpose” because the specification later describes that each microprocessor can a “special purpose” microprocessor. Ex.1001, 7:25-27; Ex.1163, ¶20.

B. Patent Owner excludes hardware processor configurations described in the specification

Patent Owner argues that the claim language refers to “a single processor package containing multiple cores” and “requires a single multi-core processor.” POR, 11, 15. Patent Owner, however, fails to provide an express construction. *See generally*, POR, 10-16. At best, Patent Owner provides:

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.

POR, 15. Note, though, that Patent Owner and its expert are addressing the “ordinary usage of the term “**multi-core processor**” – which is not the claim term at issue. *See id.*; Ex.2017, ¶¶26, 33. Moreover, the claims simply recite “processor,” whereas the POR conflates “processor” with “microprocessor.” POR, 13 (“one processor package”),15 (“single processor package”).

Next, Patent Owner argues that the claim language “requires a **single multi-core** processor” (intimating that a “processor” is the same as a “microprocessor”). POR, 11. This directly contradicts the specification. The specification contemplates a “processor” that “can be one or more single-core or

multi-core microprocessors.” See Ex.1001, 5:11-37, 7:14-36, 9:50-10:3. As such, the specification provides that a processor can be multiple configurations of microprocessors, such as (i) “one or more” microprocessors and (ii) “single-core or multi-core” microprocessors. Ex.1163, ¶24.

Thus, the specification’s language of a “processor ... can be one or more single-core or multi-core microprocessors” contemplates:

- (i) a processor can be one single-core microprocessor,
- (ii) a processor can be multiple single-core microprocessors,
- (iii) a processor can be one multi-core microprocessor, and
- (iv) a processor can be multiple multi-core microprocessors.

See Ex.1001, 5:11-37, 7:14-36, 9:50-10:3; Ex.1163, ¶25.

The claim language, which requires a plurality of processing cores, excludes processor configuration (i). But Patent Owner’s arguments limiting a hardware processor to only the **single multi-core** microprocessor configuration exclude processor configurations (ii) and (iv). See *Verizon v. Vonage Holdings*, 503 F.3d 1295, 1305 (Fed.Cir.2007) (rejecting proposed claim interpretation that would exclude disclosed examples in the specification); Ex.1163, ¶26. And while Patent Owner argues that the “Intel Core 2 Duo” example is “dispositive,” it is merely one example from a list of 16 examples. POR, 13; see Ex.1001, 7:14-36. See *Phillips v. AWH Corp.*, 415 F.3d 1303, 1323 (Fed. Cir. 2005) (warning against construing

claims as being limited to a “single embodiment” described in the specification).

Further, Patent Owner's reliance on the “Intel Core 2 Duo” ignores another example processor from the list: the “MIPS processor.” Ex.1001, 7:24, 9:59. MIPS processors were known by POSITAs to be a “multiprocessor” composed of a plurality of individual microprocessors (called “coprocessors”). *See* Ex.1168, 14, 20 (“The MIPS Architecture defines four coprocessors” with “uniprocessor and multiprocessor (MP) implementations”). Accordingly, the specification not only contemplates a processor configuration composed of multiple microprocessors, it lists the MIPS processor as an example of one. Ex.1163, ¶27.

Thus, Patent Owner's arguments contradict the specification, while excluding embodiments expressly described in the specification from the scope of independent claims 1 and 30. Patent Owner's arguments should be rejected. Ex.1163, ¶¶24-27.

C. The “plurality of processing cores” language does not limit the “hardware processor” to only a “multi-core microprocessor”

The Board also asked the parties to address the specification's description that “processor cores are generally independent microprocessors, but more than one can be included in a single chip package.” Ex.1001, 5:14-16 (cited at DI, 14). This description supports construing “hardware processor with a plurality of processing cores” to encompass a processor configuration of “one or more”

microprocessors, because the “plurality of processing cores” would “generally” be a plurality of “**independent microprocessors**.” Ex.1001, 5:14-16; Ex.1163, ¶28.

The POR characterizes this disclosure as “expressly locat[ing] multiple cores within a ‘single-chip package.’” POR, 12. However, the specification states that “more than one [processing core] **can be** included in a single chip package,” and does *not require* them to be in the same package. Ex.1001, 5:14-16. The specification positions the single chip package configuration as another example alongside its description that “processor cores are generally independent microprocessors.” Ex.1001, 5:14-16. The POR attempts to avoid this disclosure by characterizing it as “explain[ing] that a ‘core’ can be microprocessor-like in capability.” The specification, however, does not mince words, clearly stating that “**cores are generally independent microprocessors**.” Ex.1001, 5:14-16; Ex.1163, ¶29.

D. The Board should reject Patent Owner's arguments

As described above, the specification states that a “**processor**” (i.e., a “hardware processor”) can include “**one or more...single-core...microprocessors**” and that “processor cores are generally **independent microprocessors**.” Ex.1001, 5:14-16, 7:16-18. Patent Owner limits the “processor” (“hardware processor”) to a “single multi-core processor” where “more than one” processing core is “included in a single chip package.” POR, 13-

14; Ex.1001, 5:14-16, 7:16-18. The claims do not recite “single chip package” and there is basis to exclude a processor configured with “multiple single-core” microprocessors and a processor configured with “multiple multi-core” microprocessors described in the specification. Accordingly, the Board should reject Patent Owner's construction. Ex.1163, ¶¶30-32.

III. THE PETITION ESTABLISHES THAT THE CHALLENGED CLAIMS ARE OBVIOUS

A. The “Symmetric MultiProcessor (SMP)” in RS6000 teaches a “*hardware processor with a plurality of processing cores*”

1. RS6000 teaches a processor having multiple microprocessors each having at least one core.

As discussed above (*see* §II), the specification describes a “processor” having multiple “single-core...microprocessors,” and describes that “processor cores are generally independent microprocessors.” Ex.1001, 5:14-16, 7:16-18; Ex.1163, ¶33.

Similarly, RS6000 teaches that each node has a “Symmetric MultiProcessor” (“hardware processor”) that includes “four, eight, twelve, or sixteen 375 MHz 630FP 64-bit processors” (plurality of processing cores). Ex.1007, 2, 9; Ex.1163, ¶34.

When the language “hardware processor with a plurality of processing cores” is properly construed in light of the specification, it encompasses a

processor having multiple single-core microprocessors—such as the Symmetric MultiProcessor (SMP) described in the RS6000 reference and applied in the Petition. Ex.1001, 5:11-37, 7:14-36, 9:50-10:3; Petition, 42-43; Ex.1007, 2, 9; Ex.1163, ¶¶35-37. In the POR, Patent Owner does not address the “one or more ... microprocessors” language of the specification as applied to the teachings of RS6000.

2. Even applying Patent Owner's arguments, RS6000 teaches a single chip package (Power3) having 8 cores.

Patent Owner interprets the Challenged Claims to “require a single hardware processor that includes multiple processor cores” in a “single chip package.” POR, 17. Even when applying Patent Owner's interpretation, the RS6000 renders obvious a “multi-core processor” in a “single chip package.” Ex.1163, ¶38.

RS6000 describes that the “Symmetric MultiProcessor (SMP)” can include multiple “POWER3” microprocessors. Ex.1007, 1-2, 9-12; Ex.1010, 52. Each POWER3 microprocessor “is a single chip implemented with 0.25 micron CMOS technology” that is “comprised of **eight execution units**...allow[ing] **concurrent operation** of ... instructions.” Ex.1010, 57. Notably, the specification also describes “processing cores” as “**units capable** of single-threaded **execution**[.]” Ex.1001, 7:27-29. Accordingly, the single POWER3 chip has “eight execution units,” and thus teaches a single chip package having 8 processing cores (which are

defined by the specification as “units capable of ... execution”). Ex.1163, ¶¶39-40; Ex.1010, 57; Ex.1001, 7:27-29. Accordingly, even under Patent Owner's improperly narrow construction, RS6000's disclosure of a POWER3 microprocessor having eight execution units renders obvious the claimed *hardware processor with a plurality of processing cores*. Ex.1163, ¶41; Ex.1010.

B. Menon/Trefethen teaches the “1→2→3→1” order of operations in the Challenged Claims

1. Menon/Trefethen's MultiMATLAB system performs the claimed order of operations

The Challenged Claims recite multiple data passing operations between nodes (e.g., limitations [1.4.2], [1.5.2], and [1.6]) that Patent Owner refers to as the “1→2→3→1 Order of Operations.” POR, 34-35; Petition, 44-48. These limitations recite that the “first node” calls the “second node” (“1→2,” limitation [1.4.2]), the “second node” performs a calculation and communicates a result to the “third node” (“2→3,” limitation [1.4.2]), and the “third node” performs a calculation using the received results, and communicates a result of that calculation to the “first node” (“3→1,” limitations [1.5.2], [1.6]). Ex.1001, claim 1; Ex.1163, ¶42; Petition, 44-48; POR, 34-35. The Menon-Trefethen combination teaches these limitations, and thus the “1→2→3→1” order of operations. Ex.1005, 6, FIG. 1; Ex.1006, 2, 4, 6-7; Petition, 44-49; Ex.1003, ¶¶211-228.

As described in the Petition, Menon teaches a “MultiMATLAB” system with interconnected nodes (“first node,” a “second node,” and a “third node” as shown in Menon’s FIG. 1):

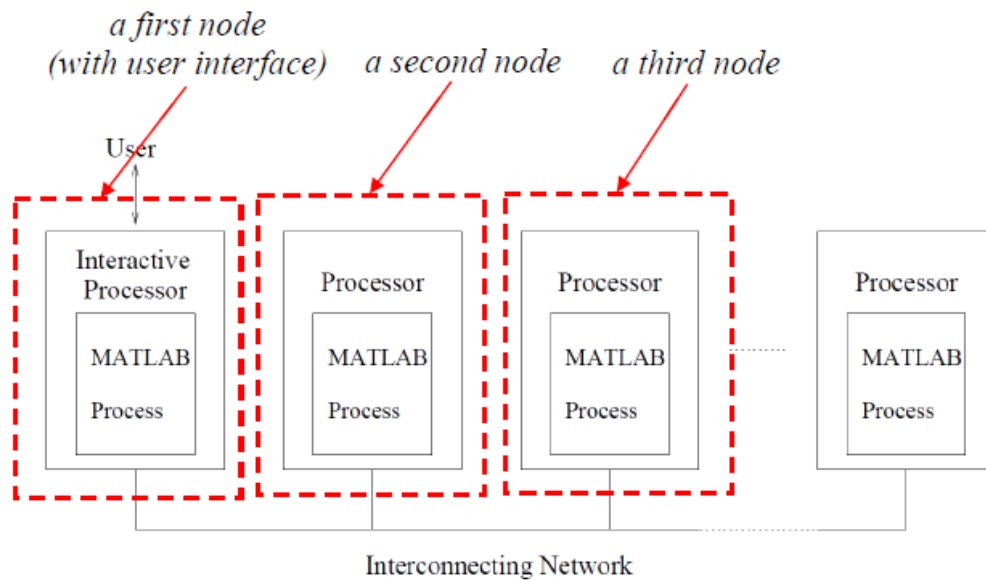


Figure 1: MultiMATLAB Architecture

Ex.1005, 3, 6, FIG. 1 (annotated); Petition, 39-46; Ex.1003, ¶217; Ex.1163, ¶43

Trefethen provides example code describing the sequence of operations that MultiMATLAB’s interconnected nodes perform:

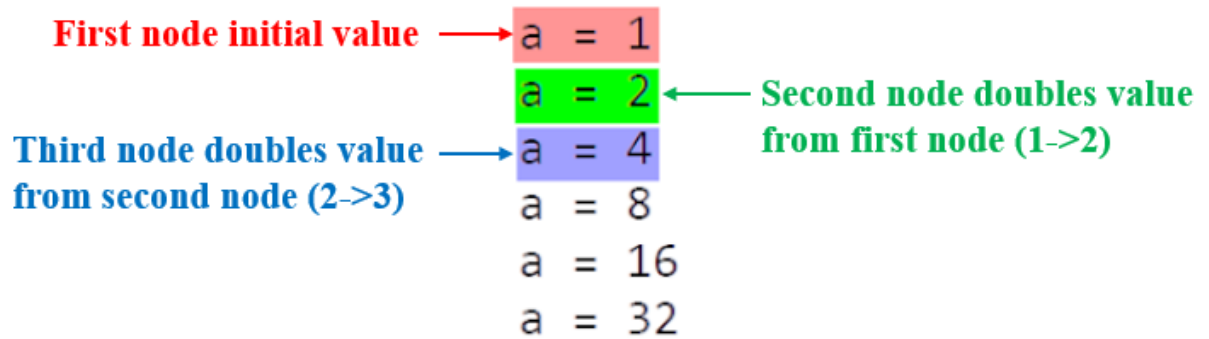
```
if ID==0                % first process: send
    a = 1
    Send(ID+1,a)
elseif ID == Nproc-1    % last process: receive and double
    a = 2*Recv
else                    % middle processes: receive, double, and send
    a = 2*Recv
    Send(ID+1,a)
end;
```

Process 0 creates the variable *a* with value 1 and sends it to process 1. Process 1 receives the message, doubles the value of *a*, and sends it along to process 2; and so on. If there are six processors the command Eval('cycle') produces the output

```
a = 1
a = 2
a = 4
a = 8
a = 16
a = 32
```

Ex.1006, 6 (annotated); Ex.1003, ¶190; Petition, 38; Ex.1163, ¶44

As described above, (i) Process 0 (i.e., the first node) “creates the variable *a* with a value 1” and sends the variable to process 1 (i.e., the second node), then (ii) process 1 (second node) “receives the message” from the process 0 (first node) and “doubles the value of *a*” (now *a*=2), and “sends it along” to process 2 (i.e., the third node). Ex.1006, 6; Petition, 21; Ex.1003, ¶114. Trefethen further describes that each MATLAB process transmits its output “to the master process” (i.e., the first node) “as soon as it is ready.” Ex.1006, 4; Petition, 47; Ex.1003, ¶221. When this sequence is executed in an example MULTIMATLAB system including six nodes, the user interface of the first node produces the following output:



Ex.1006, 6 (annotated); Ex.1163, ¶45; Petition, 47; Ex.1003, ¶220

As shown, the value of a at the first node is 1 (“a=1” in first line). Ex.1006, 6; Petition, 44-45; Ex.1003, ¶¶211-215. The value of a from the first node (1) is doubled by the second node to 2 (“a=2” in second line). Ex.1006, 6; Petition, 45; Ex.1003, ¶¶211-215. The value of a from the second node (2) is doubled again by the third node to 4 (“a=4” in third line). Ex.1006, 6; Petition, 46-47; Ex.1003, ¶¶218-220. And that value of a from the third node (“a=4”) is sent to the first node and displayed to the user. Ex.1006, 6-7; Petition, 48-49; Ex.1003, ¶¶226-228; Ex.1163, ¶46.

Accordingly, the Petition establishes that MultiMATLAB described in Menon and Trefethen teaches the “1→2→3→1” order of operations recited in the Challenged Claims. Ex.1005, 6, FIG. 1; Ex.1006, 2, 4, 6-7; Petition, 44-49; Ex.1003, ¶¶211-228; Ex.1163, ¶¶47-48.

2. Patent Owner's attempts to distinguish Menon/Trefethen fail

Patent Owner argues about multiple supposed distinctions between the Challenged Claims and the "1→2→3→1" order of operations taught by Menon/Trefethen. POR, 35-40.

(a) "Master-controlled dispatch model" is not a distinction

Patent Owner characterizes Menon/Trefethen's architecture as a "master-controlled dispatch model" in which "a user interacts with an 'interactive process'" (i.e., the first node) "and **other processes**" (i.e., the second and third nodes) "**wait for commands from that interactive process.**" POR, 35; Ex.1163, ¶49.

First, Patent Owner's argument about Menon/Trefethen ignores that the claims themselves *require* a node that receives user instructions while other nodes wait. For example, each of claims 1 and 30 recite a node that receives user instructions while the other nodes wait: "*kernel configured to interpret user instructions and distribute calls to at least one of the plurality of other nodes for execution.*" Ex.1001, 29:47-51; 34: 33-35; Ex.1163, ¶¶49-51.

Second, Patent Owner argues that Trefethen's script "passes a value from process to process using hardcoded send/receive instructions." POR, 36. This, of itself, acknowledges that Menon/Trefethen teaches passing a value from process to

process in accordance with instructions. As discussed above, Trefethen's script describes:

Process 0 creates the variable *a* with value 1 and sends it to process 1. Process 1 receives the message, doubles the value of *a*, and sends it along to process 2; and so on. If there are six processors the command `Eval( 'cycle'`
`)` produces the output

```
a = 1  
a = 2  
a = 4  
a = 8  
a = 16  
a = 32
```

Ex.1006, 6; Petition, 21; Ex.1163, ¶52.

Correspondingly, the claims require that the second node (Trefethen's "process 1") receives a call from the first node ("*Process 0* creates the variable *a* with value 1 and *sends it to process 1*"), performs a mathematical operation (process 1 "doubles the value of *a*") and communicate the result to the third node (process 1 "sends it along to process 2"), which is configured to receive the call from the second node and then performs a mathematical operation on the result received from the second node ("and so on"), resulting a value of "*a* = 4." Ex.1006, 6; Ex.1163, ¶53.

(b) The Challenged Claims do not exclude "user-authored application code"

Next, Patent Owner attempts to explain away the "1→2→3→1" order of operations demonstrated by Menon/Trefethen as "simply an example of user-authored application code that passes a value from process to process using

hardcoded send/receive instructions.” POR, 36. Again, Patent Owner contradicts the specification. The specification describes that the kernel interprets “code” and “instructions” supplied by a “user or a script.” Ex.1001, 23: 1-5. Similarly, Menon/Trefethen describe MATLAB interpreting code and instructions supplied by user commands and scripts. Ex.1005, 14 (“this MATLAB code”); Ex.1006, 4-5 (examples of user commands), 5-6 (example “m-file”); *see also* Ex.1016, 5 (m-files are “scripts”); Ex.1163, ¶¶54-55. Patent Owner fails to identify any distinction between “user-authored application code” and the “scripts” as described in the specification, much less the Challenged Claims. POR, 36-37.

Moreover, at the District Court, in direct contradiction to its arguments in this proceeding, Patent Owner argued that user-authored application code is within the scope of the claims. When alleging infringement, Patent Owner argued:

Intel provides ‘HCCL Demo,’ which is ‘a program utilizing HCCL APIs, demonstrating both collective and point-to-point communications.’ The source code for HCCL Demo includes the details ... for distributing calls to at least one of a plurality of ... nodes, for receiving the result of mathematical expression from other nodes, for executing mathematical expression evaluations, and for communicating a result of mathematical expression evaluation to other nodes.

Ex.1022, 46-47.

As such, Patent Owner has taken the position that a user-authored application code's source code that distributes calls, receives results, evaluates mathematical expressions and communicates results is within the scope of the claims. *See Southwall Techs., Inc. v. Cardinal IG Co.*, 54 F.3d 1570, 1576 (claims are not "construed one way in order to obtain their allowance and in a different way against accused infringers").

Accordingly, the Challenged Claims should not be treated to exclude "user-authored application code" and thus the process described in Menon/Trefethen teaches the "1→2→3→1" order of operations of the Challenged Claims. Petition, 44-49; Ex.1003, ¶¶211-228; *see* §IV.B.1; Ex.1163, ¶¶54-56.

(c) The Challenged Claims do not require distinct "mathematical expressions" to be performed at each node

Patent Owner argues that the Challenged Claims require "two **distinct** mathematical expression evaluations executed at two different nodes," and therefore Menon/Trefethen's teaching of the "doubling operation" executed by the second node on the value received from the first node ("a first mathematical expression evaluation") and the "doubling operation" executed by the third node on the result received from the second node ("a second mathematical expression evaluation") do not teach the claimed features. POR, 37.

First, this is incorrect. Trefethen's example code teaches that process 1's expression evaluation is "1*2," while process 2's expression evaluation is "2*2." *See* Ex.1006, 6. The different nodes of Trefethen are performing distinct mathematical expression evaluations. *See* Ex.1006, 6; Ex.1163, ¶¶57-58, 61.

Second, Patent Owner's arguments conflict with the claim language. The Challenged Claims recite a first and second "mathematical expression **evaluation**," and that each node is performing *its own* "evaluation" of a mathematical expression. In other words, an "evaluation" of a mathematical expression by one node is necessarily distinct from an "evaluation" of the same mathematical expression by another node. *See, e.g.*, Ex.1001, 14:34-35. This is precisely what Trefethen's example code describes. *See* Ex.1006, 6; Ex.1163, ¶¶59, 61.

Third, Patent Owner again ignores the language of the specification. For example, the specification describes that each node can perform the "same operations." Ex.1001, 16:42-17:10 (describing how the cluster node modules perform "Parallel Evaluation" using "routines" that "perform **the same operations**"). Each of Trefethen's nodes are performing that same *operation* (e.g., doubling). *See* Ex.1006, 6; Ex.1163, ¶¶60.

Thus, Patent Owner's argument fails.

(d) The Challenged Claims do not have a distinction between "screen output" and "data"

The Challenged Claims recite that “the third node is configured to communicate the result of the second mathematical expression evaluation to the first node” (limitation [1.5.2]) and “the first node is configured to return the result of the second mathematical expression evaluation to the user interface” (limitation [1.6]). As discussed above, Menon/Trefethen render obvious these limitations. In Menon/Trefethen, a result from the third node (i.e., “a=4”) is communicated to the first node and displayed to the user in a user interface of the first node. Petition, 46-49; Ex.1003, ¶¶218-228; see §IV.B.1.

Result from third node displayed in the user interface of the first node →

a =	1
a =	2
a =	4
a =	8
a =	16
a =	32

Ex.1006, 6 (annotated); Ex.1163, ¶¶62-63; Petition, 47; Ex.1003, ¶220.

Patent Owner argues, again without citing any evidentiary support, that the Challenged Claims require a “mechanism by which the value computed on a remote node is returned to the originating process as **data for further use.**” First, Patent Owner acknowledges that Menon/Trefethen teaches that the first node receives the result and *then displays it*. See POR, 27 (“cause the remote node to

receive and thereby print the values”). In other words, the first node *does* put the result to “further use” after receiving it: it displays it on the screen for the user.

Ex.1163, ¶64.

Second, to the extent Patent Owner is arguing that the claims require some specific action or use of the result, the claims do not recite any such “further use.” In fact, the drafter of the Challenged Claims knew how to claim “further use” (e.g., the third node performs a mathematical evaluation using the value received from the second node), yet the claims contain no such language requiring any “further use” after the result is received by the first node. Ex.1163, ¶65.

Patent Owner also attempts to distinguish Menon/Trefethen by arguing that the display of that result by the first node in Menon/Trefethen is merely “forwarded...screen output,” and not “data.” POR, 24-31. This is incorrect. The values displayed on the screen *are* data. Ex.1006, Abstract (MultiMATLAB uses “Eval, Send, [and] Recv” to “pass commands and **data** between [] these various [MATLAB] processes.”). Further, the Challenged Claims (e.g., limitations [1.5.2] and [1.6], reproduced above) recite no requirement that the “*result of the second mathematical expression evaluation*” communicated by the third node to the first node must be “data” as Patent Owner argues. Ex.1001, claim 1. Nor does the claim language require any particular “mechanism” for the communication of the claimed result that would exclude communicating screen output that includes the

computed result from the third node to the first node, as described in

Menon/Trefethen. Ex.1001, claim 1; Ex.1163, ¶¶66-69.

Accordingly, Patent Owner's arguments seek to improperly narrow the scope of the Challenged Claims without explanation or supporting evidence and thus fail.

C. The Petition establishes that combining Menon and Trefethen would have been obvious

1. Menon and Trefethen teach how to implement MultiMATLAB

The Petition, supported by Dr. Bajaj's expert testimony (Ex.1003, ¶¶65-73), describes multiple reasons why a POSITA would have been motivated to combine Menon and Trefethen, including (Ex.1163, ¶¶70-72):

- Menon suggesting the combination by **expressly citing** Trefethen and explaining that Menon builds on the teachings of Trefethen. Ex.1005, 1-2; Petition, 22, Ex.1003, ¶71.
- Menon and Trefethen both describe aspects of how the same MultiMATLAB system enables MATLAB to perform parallel mathematical calculations. Ex.1005, Abstract, 16; Ex.1006, 2; Petition, 22-23; Ex.1003, ¶72.
- Menon and Trefethen together represent a combination of prior art elements according to known methods (executing MATLAB code on a multi-node system) to yield the predictable result discussed in both references: the

MultiMATLAB system executing MATLAB in parallel to perform mathematical calculations. Ex.1005, Abstract; Ex.1006, 2-7; Petition, 23; Ex.1003, ¶70.

POR ignores the articulated reasons to combine and instead reduces Petitioner's reasons to combine to "researchers had a strong desire to parallelize MATLAB" (which it labels as "Petitioner's motivation"). POR, 41 (quoting Petition, 23).⁶ And Patent Owner's position is entirely attorney argument with no supporting expert testimony on this issue.

2. Menon does not teach away from the combination

The Menon and Trefethen references share two of the same authors—"Vijay Menon" and "Anne E. Trefethen" are the first two authors listed on both papers. Petition, 23; Ex.1005, 1; Ex.1006, 1. Both references describe aspects of the same system—"MultiMATLAB." Petition, 22-23; Ex.1005, 1 (titled "MultiMATLAB: Integrating MATLAB with High-Performance Parallel Computing"); Ex.1006, 1 (titled "MultiMATLAB: MATLAB on Multiple Processors"). Menon explicitly

⁶ The POR indicates this quote is from page 21 in the "Motivation to combine" section of the Petition. The actual quote appears on page 23 in the "Predictable and reasonable expectation of success" section.

cites Trefethen and explains that Menon builds on the teachings of Trefethen.

Ex.1005, 1-2; Ex.1003, ¶71; Petition, 22.

Despite these facts, the POR argues that Menon “teaches away” from Trefethen because Menon describes the “MultiMATLAB architecture” of Trefethen as “different from the newer one described in” Menon. POR, 42-44. But Menon does not “distinguish” the “MultiMATLAB” teachings of Trefethen as the POR argues. In fact, Menon describes *enhancing* the system of Trefethen. Ex.1005, 2. For example, Menon lists the “goals” already achieved by MultiMATLAB and then states that the architecture has been “redesigned to achieve two **further** goals.” Ex.1005, 2. One of these two “further goals” is “extensibility,” which Menon achieves by adding “MEX routines” to the system of Trefethen. Ex.1005, 2-3. Far from showing an “incompatibility” between Menon and Trefethen as the POR argues, Menon’s addition of “MEX routines” to MultiMATLAB shows that such a modification was within the capabilities of a POSITA—in fact, the authors of the Menon paper *actually implemented* the modification. Ex.1163, ¶¶73-74; Ex.1005, 2-3.

A POSITA reading Menon would not have been led to disregard the teachings of Trefethen, something Menon itself does not do. Ex.1005, 1-2 (citing Trefethen); *DePuy Spine v. Medtronic*, 567 F. 3d 1314, 1327 (a reference teaches away if a POSITA “would be discouraged from following the path set out in the

reference”). Accordingly, Menon does not teach away from the combination with Trefethen described in the Petition, and Patent Owner’s arguments fail. Ex.1163, ¶¶75-77.

IV. THE REFERENCES CITED IN THE PETITION ARE PRINTED PUBLICATIONS

The Petition, supported by Dr. Bajaj’s testimony, establishes that Menon, Trefethen, RS6000, POEref, and MPIRef qualify as printed publications. Petition, 14-16. The POR arguments on this issue are like those presented in Patent Owner’s Preliminary Response (Paper 10), which the Board found unpersuasive in its Institution Decision. *See* DI, 39-49. The POR, however, ignores the Decision’s lengthy discussion of the shortcomings in Patent Owner’s arguments and, instead, repeats the same flawed arguments without supporting evidence or expert testimony. POR, 58-66.

1. Menon is a printed publication

Menon qualifies as a printed publication because it was published and distributed at the Proceedings of the 1997 ACM/IEEE Conference on Supercomputing 1997. Ex.1005; Ex.1026; Ex.1027. Menon was also published and distributed on Cornell’s website at least as early as 2001. Ex.1023; Ex.1029. Further, Menon was sufficiently disseminated and was located by POSITAs. *See*

Suffolk Techs., LLC v. AOL Inc., 752 F.3d 1358, 1365 (Fed. Cir. 2014). For example, the following interested persons cited to Menon (Ex.1086, ¶¶54-63):

- Ex.1094, 16 (note 14) (published in 1998, citing “V. Menon and A. Trefethen, MultiMATLAB: Integrating MATLAB with High-Performance Parallel Computing, Proceedings of Supercomputing '97 (SC97)”);
- Ex.1095, 223 (note 2) (published in 1999, citing “Vijay Menon and Anne E. Trefethen, ‘MultiMATLAB: Integrating MATLAB with High-Performance Parallel Computing’, SuperComputing97, 1997”);
- Ex.1096, 13 (note 17) (published in 2000, citing “V. Menon and A. E. Trefethen, MultiMATLAB: Integrating MATLAB with high-performance parallel computing. In *Proceedings of Supercomputing '97*, 1997”); and
- Ex.1097, 72 (published in 2000, citing “[Menon and Trefethen, 1997] Menon, V. and Trefethen, A. (1997). MultiMATLAB: Integrating MATLAB with high-performance parallel computing. In Proc. Supercomputing '97”).

Additionally, Menon was indexed and cataloged in library records:

- “Library records in WorldCat.org” (Ex.1090) showing Menon was available in “the German National Library of Science and Technology” as of “February 19, 1998.” Ex.1086, ¶¶33-36;
- “MARC records” (Ex.1091) showing Menon was cataloged in the Academia Sinica Library on “December 14, 1998.” Ex.1086, ¶¶37-43; and
- “Internet Archive records” (Ex.1092) showing a copy of Menon was archived on “July 7, 1998.” Ex.1086, ¶¶44-53.

Accordingly, Menon was cataloged, indexed, and read and used by POSITAs. Menon qualifies as a printed publication. Petition, 14 (citing Ex.1005; Ex.1026; Ex.1027; Ex.1028; Ex.1003, ¶¶110-112); Ex.1086, ¶¶28-70.

2. Trefethen is a printed publication

Trefethen qualifies as a printed publication because it was published and distributed on the webpage of one of its authors at least as early as 2001. Ex.1023; Ex.1029. Trefethen was also published on a Cornell University website no later than May 10, 1996. Ex.1023; Ex.1029. Further, Trefethen was sufficiently disseminated and was located by POSITAs. *See Suffolk Techs., LLC* at 1365. For example, the following interested persons cited to Trefethen (Ex.1086, ¶¶86-99):

- Ex.1102 (published in 1997, citing to “Anne E. Trefethen, Vijay S. Menon, Chi-Chao Chang, Grzegorz J. Czajkowski, and Chris

Myers. Multimatlab: Matlab on multiple processors. URL <http://www.cs.cornell.edu/Info/People/Lnt?multimatlab.html>, May 1996.);

- Ex.1104, 7 (note 11) (published in 1998, citing to “Trefethen, A. E., Menon, V. S., Chang, C.-C., Czajkowski, G. J., Myers, C., and Trefethen, L. N., ‘MultiMATLAB: MATLAB on Multiple Processors,’ Technical Report 96-239, Cornell Theory Center, Ithaca, NY (1996)”); and
- Ex.1107, 21 (note 12) (published in 1998, citing to “Anne E. Trefethen, Vijay S. Menon, Chi-Chao Chang, Grzegorz J. Czajkowski, Chirs Myers, and Lloyd N. Trefethen. MultiMATLAB: MATLAB on multiple processors. Technical Report 239, Cornell Theory Center, 1996”).

Additionally, Trefethen was indexed and cataloged in library records:

- “MARC records” (Ex.1101) showing that Trefethen was cataloged by “Washington State University Library” on “November 3, 1997.” Ex.1086, ¶¶79-85.
- “Internet Archive” records (Ex.1100) showing that Trefethen was archived on “May 10, 1996.” Ex.1086, ¶¶76-78;

Accordingly, Trefethen was cataloged, indexed, and read and used by POSITAs. Trefethen qualifies as a printed publication. Petition, 14-15 (citing Ex.1023; Ex.1029; Ex.1035, 2-3; Ex.1003, ¶¶116-117); Ex.1086, ¶¶71-104.

3. RS6000 is a printed publication

RS6000 qualifies as a printed publication because it was catalogued and indexed in IBM's online document library at least as early as 2002. Ex.1007; Ex.1029; Ex.1038; *see also* Ex.1003, ¶¶119-120. Further, RS6000 was sufficiently disseminated and was located by POSITAs. *See Suffolk Techs., LLC* at 1365. For example, the following interested person cited to RS6000 (Ex.1086, ¶¶120-122):

- Ex.1112, 7 (published in 2002, citing "IBM Corporation. 'RS/6000 Planning Vol. 1, Hardware and Physical Environment' URL 'http://publibfp.boulder.ibm.com/epubs/pdf/da70bmst.pdf.'").

Additionally, RS6000 was indexed and cataloged in library records:

- "Internet Archive" records (Ex.1109, Ex.1110) showing that RS6000 "publicly accessible on the Internet no later than July 26, 2001." Ex.1086, ¶¶112-119.

Accordingly, RS6000 was cataloged, indexed, and read and used by POSITAs. RS6000 qualifies as a printed publication. Petition, 15 (citing Ex.1007; Ex.1029; Ex.1038; Ex.1003, ¶¶119-120); Ex.1086, ¶¶105-125.

4. POEref is a printed publication

POEref qualifies as a printed publication because it was a user manual distributed with the IBM SP2 systems and catalogued and indexed in IBM's online document library at least as early as 2001. Ex.1008, Ex. 1029, Ex.1036, Ex.1039; Ex.1047; *see also* Ex.1003, ¶¶123-125. Further, POEref was sufficiently disseminated and was located by POSITAs. *See Suffolk Techs., LLC* at 1365. For example, the following interested person cited to POEref (Ex.1086, ¶¶138-144⁷):

- Ex.1114 (published in 2004, citing “Parallel Environment (PE) for AIX V3R2.0: Operation and Use, Vol. 1. December 2001”); and
- Ex.1116 (U.S. Patent Publication from 2005, citing “Operation and Use, Volume 1. Using the Parallel Operating Environment,” Publication Number SA22-7425-01, Version 3 Release 2, Second Edition (December 2001))”).

Additionally, POEref was indexed and catalogued in library records:

- “Internet Archive” records (Ex.1113) showing that POEref was available no later than February 2002. Ex.1086, ¶¶128-137.

⁷ Dr. Hsieh-Yee's declaration refers to POEref (Ex.1008) using the short name “POE-2001.” Ex.1086, p. 61.

Accordingly, POEref was cataloged, indexed, and read and used by POSITAs. POEref qualifies as a printed publication. Petition, 15 (citing Ex.1008, 157; Ex.1029; Ex.1036, 14; Ex.1039; Ex.1047; Ex.1003, ¶¶123-125); Ex.1086, ¶¶126-148.

5. MPIref is a printed publication

MPIref qualifies as a printed publication because it was distributed to POSITAs using FTP mail servers by the University of Tennessee and Oak Ridge National Library in 1994 and was also made available on an Argonne National Laboratory website in 1997. Ex.1017, cover page; Ex.1029; Ex.1003, ¶129. It was published on MPI Forum's website no later than July 3, 1998. Ex.1029, 1-3. Further, MPIref was sufficiently disseminated and was located by POSITAs. *See Suffolk Techs., LLC* at 1365. For example, the following interested persons cited to MPIref (Ex.1086, ¶¶167-180.⁸):

- Ex.1122 (published in 1994, citing to “MPI: A Message-Passing Interface Standard, May 1994”);

⁸ Dr. Hsieh-Yee's declaration refers to MPIref (Ex.1017) using the short name “MPI-1994.” Ex.1086, p. 71

- Ex.1124 (note 19) (published in 1995, citing to “MPI: A Message Passing Interface Standard. Tech. rep., Message Passing Interface Forum, May 1994”); and
- Ex.1126, 8 (note 11) (published in 1995, citing to “‘MPI: A Message-Passing Interface Standard,’ Technical Report CS-94-230, University of Tennessee, Knoxville, TN, May 5 1994”); and
- Ex.1128, 8 (note 7) (published in 1996, citing “MPI Committee. MPIL a message-passing interface standard. Internat. J. Supercomputer Applications, 8(3/4): 165-416, Fall/Winter 1994”).

Additionally, MPIref was indexed and catalogued in library records:

- “MARC records” (Ex.1120, Ex.1121) showing that MPIref was cataloged by the “National Science Library of Canada” on “November 22, 1994,” and by the “University of Edinburgh” on “March 30, 2004.” Ex.1086, ¶¶157-161; and
- “Internet Archive” records (Ex.1118) showing that MPIref “was publicly accessible on the Internet no later than July 3, 1998.” Ex.1086, ¶¶153-154.

Accordingly, MPIref was cataloged, indexed, and read and used by POSITAs. MPIref qualifies as a printed publication. Petition, 15-16 (citing Ex.1029; Ex.1037, 2-16; Ex.1003, ¶¶129-130); Ex.1086, ¶¶149-182.

V. THE PRIOR ART REFERENCES CITED IN THE PETITION ARE ADMISSIBLE

The prior art references cited in the Petition are authentic, and thus admissible, at least because each reference qualifies as an ancient document under Fed. R. Evid. 901(8) and is *prima facie* authenticated. Petition, 16.

To the extent Patent Owner seeks to address any issues related to admissibility, Petitioner will address any such issues in its Opposition to the Motion to Exclude. Trial Practice Guide, ¶II.K.

VI. CONCLUSION

For the foregoing reasons, Petitioner requests that all challenged claims be found unpatentable.

Respectfully submitted,

Dated: May 1, 2026
HAYNES AND BOONE, LLP
2801 N. Harwood Street, Suite 2300
Dallas, Texas 75201
Customer No. 27683

/Andrew S. Ehmke/
Andrew S. Ehmke
Lead Counsel for Petitioner
Registration No. 50,271

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), Petitioner hereby certifies, in accordance with and reliance on the word count provided by the word-processing system used to prepare this Reply, that the number of words in this paper is 5,598. Pursuant to 37 C.F.R. § 42.24(d), this word count excludes the table of contents, table of authorities, mandatory notices under § 42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing.

Dated: May 1, 2026

/Andrew S. Ehmke/

Andrew S. Ehmke
Lead Counsel of Petitioner
Registration No. 50,271

CERTIFICATE OF SERVICE

The undersigned certifies, under 37 C.F.R. § 42.6, that service was made on the Patent Owner as detailed below.

Date of service May 1, 2026

Manner of service Electronic Email: dlindner@crowell.com;
jgurka@crowell.com; rey@bhiplaw.com

Documents served **PETITIONER'S REPLY TO PATENT OWNER'S
RESPONSE; Exs.1086-1161, 1163, 1168**

Persons served David Lindner
Crowell & Moring, LLP
200 N. LaSalle Drive, Ste. 2500
Chicago, IL 60654

Jon Gurka
Crowell & Moring, LLP.
3 Park Plaza, Floor 20
Irvine, CA 92614

Reynaldo C. Barcelo
Barcelo, Harrison & Walker, LLP
2901 W. Coast Hwy, Ste. 200
Newport Beach, CA 92633

/Andrew S. Ehmke/
Andrew S. Ehmke
Lead Counsel for Petitioner
Registration No. 50,271