UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMERICAN AIRLINES, INC. AND SOUTHWEST AIRLINES CO.,

Petitioners

v.

INTELLECTUAL VENTURES I LLC,

Patent Owner.

Inter Partes Review No. IPR20254-00785
U.S. Patent No. 7,257,582 B2
Issue Date: August 14, 2007

Title: LOAD BALANCING AND SHARED DATA

PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,257,582 B2

Mail Stop Patent Board
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

I. Mandatory Notices Under 37 C.F.R. § 42.8 2

A. Real party in interest under § 42.8(b)(1) 2

B. Related matters under § 42.8(b)(2) 3

C. Lead and backup counsel under § 42.8(b)(3) 3

D. Service information 4

II. Fee Payment 5

III. Grounds for Standing 5

IV. Brief Overview of the '582 Patent 5

V. Priority Date 6

VI. Claim Construction 6

VII. Person of Ordinary Skill in the Art (POSITA) 7

VIII. Precise Relief Requested and Grounds Raised 7

IX. Brief Description of the Applied Prior Art References 9

A. Chow (Ex.1003) 9

B. Reiffin (Ex.1004) 9

C. Kurowski (Ex.1005) 10

X. The Challenged Claims are Unpatentable 10

A. Ground 1: Claims 1-14 are obvious over Chow in view of Reiffin 10

1. Claim 1 12

2. Claim 2: The method defined in claim 1 wherein the automatic determination of file allocation and logical subdivision of

records of said input file into said plurality of partitions in step (a) and the distribution of the description of all of said partitions in step (b) is carried out with at least one further processor in addition to the subtask processors. 20

3. Claim 3: The method defined in claim 1, further comprising the step of merging said subtask outputs in step (e). 21

4. Claim 4: The method defined in claim 1 wherein the output in step (e) is a succession of outputs from said subtasks in a one-to-one correspondence with said records of said input file. 22

5. Claim 5: The method defined in claim 1 wherein the output in step (e) is an accumulation of output records from said subtasks in an arbitrary order. 23

6. Claim 6: The method defined in claim 1 wherein said input file resides on a storage area network and is derived therefrom. 24

7. Claim 7: The method defined in claim 1 wherein said input file resides on a network-attached storage and is derived therefrom. 25

8. Claim 8: The method defined in claim 1 wherein said computer-executable process is a sort process. 26

9. Claim 9: The method defined in claim 1 wherein said computer-executable process is a statistical analysis process. 27

10. Claim 10: The method defined in claim 1 wherein said computer-executable process is a report-creating process. 28

11. Claim 11: The method defined in claim 1 wherein said computer-executable process includes a database query.....29

12. Claim 12: The method defined in claim 2 wherein said one processor is part of a mainframe computer and the other processors are processors of at least one other computer. 30

13. Claim 13: The method defined in claim 1 wherein said plurality of processors are all parts of a single multiprocessor. 32

14. Claim 14: The method defined in claim 1 wherein the automatic determination of file allocation and logical subdivision of records of said input file into said plurality of partitions in step (a) and the distribution of the descriptions of all of said partitions in step (b) is carried out with at least one processor, and said one processor and said plurality of processors are all parts of a single multiprocessor not including said one processor. 33

B. Ground 2: Claims 1-14 are obvious over Chow in view of Reiffin and further in view of Kurowski.....34

1. Claim 1 37

2. Claim 2: The method defined in claim 1 wherein the automatic determination of file allocation and logical subdivision of records of said input file into said plurality of partitions in step (a) and the distribution of the description of all of said partitions in

step (b) is carried out with at least one further processor in addition to the subtask processors. 43

3. Claims 3-14 44

XI. The Board Should Institute *Inter Partes* Review 44

A. The *Fintiv* factors either favor institution or are neutral 44

B. An analysis under 35 U.S.C. § 325(d) also favors institution 46

XII. Conclusion 48

TABLE OF AUTHORITIES

Page(s)

CASES

Advanced Bionics, LLC v. Med-El Elektromedizinische Geräte GMBH,
IPR2019-01469, Paper 6 (P.T.A.B. Feb. 13, 2020).....46, 47

Becton, Dickinson & Co. v. B. Braun Melsungen AG,
IPR2017-01586, Paper 8 (P.T.A.B. Dec. 15, 2017) 45, 47

In re GPAC Inc.,
57 F.3d 1573 (Fed. Cir. 1995) 7

In re Nilssen,
851 F.2d 1401 (Fed. Cir. 1988) 8

Maplebear Inc. v. Consumeron, LLC,
IPR2022-01357, Paper 9 (P.T.A.B. Mar. 3, 2023) 47

Phillips v. AWH Corp.,
415 F.3d 1303 (Fed. Cir. 2005) 6

Sand Revolution II, LLC v. Continental Intermodal Grp.-Trucking LLC,
IPR2019-01393, Paper 24 (P.T.A.B. June 16, 2020) 44, 45

Sotera Wireless, Inc. v. Masimo Corp.,
IPR2020-01019, Paper 12 (P.T.A.B. Dec. 1, 2020) 45

Unwired Planet, LLC v. Google, Inc.,
841 F.3d 995 (Fed. Cir. 2016) 8, 9

STATUTES

35 U.S.C. § 102(a) 9

35 U.S.C. § 102(b) 9, 10

35 U.S.C. §103 .....6, 7

35 U.S.C. § 325(d) 45, 47

OTHER AUTHORITIES

37 C.F.R.§ 42.8 .....2, 3

37 C.F.R. § 42.100(b) 6

37 C.F.R. § 42.103(a).....5

37 C.F.R. § 42.15(a).....5

83 Fed. Reg. 51,340, 51,340-44 (Oct. 11, 2018) 6

LIST OF EXHIBITS

Exhibit No.	Description of Document
1001	U.S. Patent No. 7,257,582 B2 (the '582 Patent) to Rothschild (filed February 27, 2003; issued August 14, 2007)
1002	File history for the '582 Patent
1003	U.S. Patent No. 6,304,866 (Chow) to Chow et al. filed on June 27, 1997; issued October 16, 2001
1004	U.S. Patent No. 6,330,583 (Reiffin) to Reiffin filed on September 9, 1994; issued December 11, 2001
1005	U.S. Patent Application No. 2002/0019844 (Kurowski) to Kurowski et al. filed January 12, 2001; published February 14, 2002
1006	Complaint, *Intellectual Ventures I LLC et al. v. American Airlines, Inc.*, No. 4:24-cv-980-ALM (E.D. Tex. Nov. 2, 2024)
1007	Claim Chart for '582 Patent, *Intellectual Ventures I LLC et al. v. American Airline, Inc.*, No. 4:24-cv-980-ALM (E.D. Tex. Nov. 2, 2024) (Dkt. 1-14) (Hadoop)
1008	Claim Chart for '582 Patent, *Intellectual Ventures I LLC et al. v. American Airline, Inc.*, No. 4:24-cv-980-ALM (E.D. Tex. Nov. 2, 2024) (Dkt. 1-15) (Spark)
1009	"Some Thoughts on Parallel Processing," Lynn D. Yarbough, Communications of the ACM, Vol. 3, Issue 10, page 539, October 1, 1960 (available at: https://dl.acm.org/doi/10.1145/367415.367426).
1010	"A Survey of Some Theoretical Aspects of Multiprocessing," J. L. Baer, ACM Computing Surveys (CSUR), Volume 5, Issue 1 Pages 31 – 80, March 1, 1973 (available at: https://dl.acm.org/doi/10.1145/356612.356615)
1011	"Parallel Sorting Algorithms for Tightly Coupled Multiprocessors," Michael J. Quinn, Parallel Computing 6 (1988) 349-357.

Exhibit No.	Description of Document
1012	"Removing Skew Effect in Join Operation on Parallel Processors," Ron-Chung Hu, Richard R. Muntz, Computer Science Technical Report, University of California, Los Angeles, June 1989.
1013	"Parallel Architectures for Database Systems," A.R. Hurson, L.L. Miller, S.H. Pakzad, M.H. Eich, B. Shirazi, Advances in Computers, Vol. 28, pages 107-151, 1989.
1014	"Parallel Algorithms for Merging and Sorting," Narsingh Deo, Dilip SarKar, Information Sciences, Vol. 56, pages 151-161, 1991.
1015	"Parallel Algorithms for lcast Median of Squares Regression," Chong-Wei Xu and Wei-Kei Shiue, Computational Statistics & Data Analysis, Vol. 16, pages 349-362, 1993.
1016	"On the Average Running Time of Odd-Even Merge Sort," Christine Rub, Max Plank-Institute Fur Informtik, Research Report, April 1995.
1017	"Parallel Algorithms," Guy E. Blelloch and Bruce M. Maggs, a chapter in "The Computer Science and Engineering Handbook." CRC Press, 1997, ISBN: 0-8493-2909-4. Available at https://dl.acm.org/doi/pdf/10.5555/1882723.1882748.
1018	"An Approach to Parallelizing Isotonic Regression," Anthony J. Kearsley, Richard A. Tapia, Michael W.Trosset, Applied Mathematics and Parallel Computing, H. Fisher et. al. (eds.), Physica-Verlag Heidelberg (1996), pages 141-147.
1019	"Part 1: The Parallel Computing Environment, Alice E. Koniges, Morris A. Jette, and David C. Eder, in "Industrial Strength Parallel Computing," Morgan Kaufmann, (2000). ISBN 9781558605404. Available at: http://wayback.cecm.sfu.ca/PSG/book/intro.html and at https://books.google.com/books?id=mWalawBciCQC&pg=PA1&source=gbs_toc_r&cad=2#v=onepage&q&f=false.
1020	GPFS: A Parallel File System, Jason Barkes, Marcelo R. Barrios, Francis Cougard, Paul G. Crumley, Didac Marin, Hari Reddy, Theerapong Thitayanun, IBM International Technical Support

Exhibit No.	Description of Document
	Organization, April 1998.
1021	"Analysis of First-Come-First-Serve Parallel Job Scheduling," Uwe Schwiegelshohn and Ramin Yahyapour. Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (1998). Available at http://dx.doi.org/10.1145/314613.315031.
1022	Declaration of Dr. Michael Shamos regarding the invalidity of U.S. Patent No. 7,257,582 B2
1023	CV for Dr. Michael Shamos
1024	Claim Chart for '582 Patent, *Intellectual Ventures I LLC et al. v. Southwest Airlines Co.*, No. 7:24-cv-277 (W.D. Tex. Nov. 2, 2024) (Dkt. 1-13) (Hadoop)
1025	Claim Chart for '582 Patent, *Intellectual Ventures I LLC et al. v. Southwest Airlines Co.*, No. 7:24-cv-277 (W.D. Tex. Nov. 2, 2024) (Dkt. 1-14) (Spark)

In November 2024, Intellectual Ventures I LLC (IV or Patent Owner) targeted multiple companies with accusations of infringing U.S. Patent No. 7,257,582 B2 (the '582 Patent) (Ex.1001).

Petitioners American Airlines, Inc. ("American") and Southwest Airlines Co. ("Southwest") (collectively "Petitioners") respectfully request that the Board now test the '582 Patent through *inter partes* review (IPR) of claims 1-14 (the Challenged Claims), constituting all claims of the '582 Patent. The '582 Patent relates to well-known computer load balancing using multiple processors. Ex.1001 (1:14-17). The Examiner rejected Applicant's claims as obvious. *See* Ex.1002 at 38 (rejecting claims as obvious "over Kitsuregawa et al. (U.S. Patent 5,603,028) in view of Pian (U.S. Patent 5,357,632)"). Applicant overcame the Examiner's rejection only after arguing his claims were distinguishable from the prior art because, rather than implementing a "special control process that uses load information to distribute the load between processors that share the load," "the load sharing process take[s] parts of the load on a first-come/first served basis." *Id.* at 53. That is the *sole* point of novelty identified by Applicant. *Id.* at 53-54. But, given the file history and Petitioners' new reference combinations discussed below, the Board should now cancel the Challenged Claims.

First, The Examiner did not consider any of Petitioners' references, let alone whether any of those references teach the purported point of novelty. Petitioners'

references teach load sharing on a "first-come/first-served basis." Thus, the Challenged Claims are obvious.

Second, the facts strongly favor institution of this IPR. Every factor under the Board's decision in *Fintiv* either favors institution or is neutral.

At bottom, the Challenged Claims relate to a crowded technological field and were issued solely based on an alleged narrow point of novelty. The additional prior art herein demonstrates that the alleged point of novelty is not novel at all. Petitioners respectfully request the Board to institute IPR of the '582 Patent and cancel the Challenged Claims.

I. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8

A. Real party in interest under § 42.8(b)(1)

American is a real party in interest to this petition. American is a wholly owned subsidiary of American Airlines Group, Inc., which is not a real party in interest to this petition. No unnamed entity is funding, controlling, or otherwise has an opportunity to control or direct this petition or American's participation in any resulting IPR.

Southwest is also a real party in interest to this petition. No unnamed entity is funding, controlling, or otherwise has an opportunity to control or direct this petition or Southwest's participation in any resulting IPR.

B. Related matters under § 42.8(b)(2)

To the best of Petitioners' knowledge, the '582 Patent is or has been involved in these matters:

Name	No.	Court	Filed	Status
Intellectual Ventures I LLC et al. v. American Airlines, Inc.	4:24-cv-980	EDTX	Nov. 2, 2024	Active
Intellectual Ventures I LLC et al. v. Southwest Airlines Co.	7:24-cv-277	WDTX	Nov. 2, 2024	Active

C. Lead and backup counsel under § 42.8(b)(3)

Petitioners designate the following counsel:

Lead Counsel	Backup Counsel
John B. Campbell (Reg. No. 54,665) jcampbell@McKoolSmith.com MCKOOL SMITH, P.C. 303 Colorado Street, Suite 2100 Austin, TX 78701 Tel: (512) 692-8730 Fax: (512) 692-8744	Alan Block (Reg No. 35,450) ablock@mckoolsmith.com McKool Smith, P.C. 300 South Grand Avenue, Suite 2900 Los Angeles, California 90071 Tel: (213) 694-1200 Fax: (213) 694-1234 Casey Shomaker (Reg. No. 77,998) cshomaker@mckoolsmith.com McKool Smith, P.C. 300 Crescent Court, Suite 1200 Dallas, TX 75201 Tel: (214) 978-4218 Fax: (214) 978-4044
	Keith D. Harden (Reg. No. 74,472)

Lead Counsel	Backup Counsel
	kharden@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616
	S. Wallace Dunwoody (Texas Bar No. 24040838 - admission *pro hac vice* to be requested) wdunwoody@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616
	Michael C. Wilson (Texas Bar No. 21704590 - admission *pro hac vice* to be requested) mwilson@munckwilson.com Munck Wilson Mandala, LLP 2000 McKinney Ave., Ste. 1900 Dallas, Texas 75201 Tel: (972) 628-3600 Fax: (972) 628-3616

D. Service information

USPTO records show the attorneys having power of attorney over the ’582 Patent are Brundidge & Stanger, P.C. This petition is thus being served by Federal Express to the correspondence address for the ’582 Patent, BRUNDIDGE &

STANGER, P.C., 2318 Mill Road, Suite 1020, Alexandria, VA 22314. Petitioners consent to electronic service at AA_Intellectual_Ventures@mckoolsmith.com.

II. FEE PAYMENT

The Office is allowed to charge the fees specified by 37 C.F.R. §§ 42.103(a) and 42.15(a) to Deposit Account No. 50-5723.

III. GROUNDS FOR STANDING

Petitioners certify that the '582 Patent is available for IPR and that Petitioners are not barred or estopped from requesting an IPR challenging the '582 Patent's claims on the grounds identified in this petition. Petitioners filed this petition within one year of service of the original complaint against Petitioners.

IV. BRIEF OVERVIEW OF THE '582 PATENT

The '582 Patent relates to well-known load balancing using multiple computer processors. Ex.1001 (1:14-17). In other words, the '582 Patent deals with "a way to split the performance of a given task among a plurality of processing units which can all access, directly or indirectly, the input data and the devices on which the output data is to be stored." *Id.* Further, "[t]he principal object of the present invention is to enable the decomposition of a certain type of linear processes that currently use a single computer, into equivalent parallel processes that can efficiently use any number of potentially heterogeneous computers, taking

the available capacity of each of these computers into account while optimizing execution." *Id*. (1:48-54).

V. PRIORITY DATE

For purposes of this IPR, Petitioners assume the earliest effective filing date of the '582 Patent is March 13, 2002. On that date, Provisional Application No. 60/363,853 ('853 provisional) was filed. The '582 Patent claims priority to the '853 provisional. Ex.1001 (Cover). Thus, this petition applies pre-AIA 35 U.S.C. §103 in showing unpatentability. Petitioners' prior art references qualify as prior art. *See infra* Section IX.

VI. CLAIM CONSTRUCTION

Petitioners apply the *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005), claim construction standard. 83 Fed. Reg. 51,340, 51,340-44 (Oct. 11, 2018). Petitioners do not believe explicit construction is needed to resolve this petition. Still, the district court case is in its early stages, having yet to commence claim construction. Petitioners reserve the right to revisit constructions proposed in district court and, in accordance with 37 C.F.R. § 42.100(b), request leave to submit the district court's claim construction order so the Board may consider it.

VII. PERSON OF ORDINARY SKILL IN THE ART (POSITA)

Based on the relevant factors, *In re GPAC Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995), a POSITA "would have had a bachelor's degree in computer science, computer engineering, electrical engineering, or related field, and at least two years of experience in computer networking and parallel computing, or a person with a master's degree in one of the foregoing and at least one year of experience in the aforementioned fields. Additional education could substitute for professional experience, and vice-versa. Ex.1022 ¶ 44.

VIII. PRECISE RELIEF REQUESTED AND GROUNDS RAISED

Petitioners request cancellation of claims 1-14 based on the following obviousness grounds:

Ground	Claims	Basis for § 103 Challenge
1	1-14	Obvious over Chow (Ex.1003) in view of Reiffin (Ex.1004)
2	1-14	Obvious over Chow (Ex.1003) in view of Reiffin (Ex.1004) and further in view of Kurowski (Ex.1005)

Petitioners also submit the Declaration of Dr. Michael Shamos (Ex.1022), a qualified expert, in support of the obviousness grounds. *See* Ex.1022 ¶¶ 1-18; Ex.1023. Grounds 1 and 2 are not cumulative of the art of record, including the art of record considered by the Examiner. *See infra* Section XI(B). During prosecution, the Examiner found that the prior art failed to disclose claim 1's

execution of unprocessed partitions on a **first-come/first-served basis**, which the Applicant cited as the '582 Patent's point of novelty. *Id.* Grounds 1 and 2 address this purported deficiency.

Each of the asserted references is analogous art and is usable in an obviousness analysis. *Unwired Planet, LLC v. Google, Inc.*, 841 F.3d 995, 1000-01 (Fed. Cir. 2016). A POSITA is presumed to have been aware of such analogous art. *In re Nilssen*, 851 F.2d 1401, 1403 (Fed. Cir. 1988).

Chow, Reiffin, and Kurowski (Cited References) are in the same field of endeavor as the '582 Patent, *e.g.*, distributed computing systems and methods. *E.g.*, Ex.1001 (1:14-17) ("More specifically, [this invention] deals with a way to split the performance of a given task among a plurality of processing units which can all access, directly or indirectly, the input data and the devices on which the output data is to be stored."); Ex.1003 (1:10-11) ("The present invention relates to multiprocessing computer database systems."); Ex.1004 (1:7-14) ("This invention relates to computer networks, and more particularly, to a distributed parallel processing network system wherein a large compute-intensive task may be partitioned into subtasks which are then distributed among a plurality of personal computer or workstations for parallel execution of the subtasks in the background concurrently with the execution in the foreground of the respective local tasks of the individual workstations."); Ex.1005 ([0003]) ("The present invention relates

generally to distributed computing, and more specifically to large-scale network-distributed computing."). Thus, the Cited References are analogous art. *Unwired*, 841 F.3d at 1000.

IX. BRIEF DESCRIPTION OF THE APPLIED PRIOR ART REFERENCES

A. Chow (Ex.1003)

Chow, titled "Aggregate Job Performance in a Multiprocessing System by Incremental and On-Demand Task Allocation Among Multiple Concurrently Operating Threads," is U.S. Patent No. 6,304,866. Chow's application was filed on June 27, 1997. Ex.1003 (Cover). Thus, Chow qualifies as pre-AIA prior art. 35 U.S.C. §§ 102(a), 102(b). Chow improves distributed multi-processing systems, particularly with respect to database operations. Ex.1003 (2:42-46); Ex.1022 ¶¶ 93-98 (describing Chow).

B. Reiffin (Ex.1004)

Reiffin, titled "Computer Network of Interactive Multitasking Computers for Parallel Processing of Network Subtasks Concurrently with Local Tasks," is U.S. Patent No. 6,330,583. Reiffin's application was filed on September 9, 1994. Ex.1004. Thus, Reiffin qualifies as pre-AIA prior art. 35 U.S.C. §§ 102(a), 102(b). Reiffin improves distributed multi-processing systems wherein a large compute-

intensive task may be partitioned into subtasks for parallel execution. Ex.1004 (1:7-14); Ex.1022 ¶¶ 99-102 (describing Reiffin).

C. Kurowski (Ex.1005)

Kurowski, titled "Method and System for Network-Distributed Computing," is U.S. Patent Application No. 2002/0019844. Kurowski's application was filed on January 12, 2001, claiming priority to a provisional application filed July 6, 2000, and published on February 14, 2002. Ex.1005. Thus, Kurowski qualifies as pre-AIA prior art. 35 U.S.C. § 102(b). Kurowski improves large-scale network-distributed computing systems. Ex.1005 ([0003]); Ex.1022 ¶¶ 103-105 (describing Kurowski).

X. THE CHALLENGED CLAIMS ARE UNPATENTABLE

A POSITA would understand Chow in view of Reiffin (Ground 1) and Chow in view of Kurowski (Ground 2) to disclose or suggest every limitation of claims 1-14 for the reasons below. Because a POSITA would have been motivated to make those combinations and had a reasonable expectation of success in doing so, both grounds render claims 1-14 obvious. Ex.1022 ¶¶ 133-136; 248-252.

A. Ground 1: Claims 1-14 are obvious over Chow in view of Reiffin

Every limitation of claims 1-14 is disclosed or suggested by Chow and Reiffin. Ex.1022 ¶¶ 133-136.

Chow and Reiffin both share the '582 Patent's goal of decomposing linear computational processes that currently use a single computer into equivalent parallel processes that can use any number of heterogeneous computers. *E.g.*, Ex.1001 (1:48-54); Ex.1003 (2:42-46) ("The invention affords its users with a number of distinct advantages. Chiefly, the invention guides a multiprocessing system to perform an aggregate task as quickly as possible, with minimum workload skew among independently operating processing elements."); Ex.1004 (2:1-7) ("It is a further object of the invention to provide a system wherein a network of workstations or personal computers may operate in parallel to form a powerful multicomputer system for parallel processing of large time-consuming compute-intensive applications in a fraction of the time that it would take any of the individual computers to do the job alone."). Ex.1022 ¶ 133.

Chow is ready for improvement. Ex.1022 ¶ 134. Chow focuses on multiprocessing systems consisting of a single computer with multiple processors executing programs in parallel. *Id.* Reiffin expands this concept to multiple networked computers. *Id.* A POSITA would have recognized that benefit of combining Reiffin with Chow would be increased computational power and the ability to solve larger problems. *Id.*

Combining Chow's multi-processing system with Reiffin's teachings of a distributed parallel processing network would have been the simple combination of

known elements to obtain a predictable result and thus obvious to a POSITA. Ex.1022 ¶ 135.

It would have been routine to add the features of Reiffin to the system of Chow. Ex.1022 ¶ 136. A POSITA would have thus been motivated to combine Chow and Reiffin with a reasonable expectation of success. *Id.*

Claims 1-14 are therefore obvious over Chow in view of Reiffin.

1. Claim 1

a. [1pre] A method of effecting on a preexisting input file a computer-executable process comprised of a plurality of subtasks, the method comprising the steps of:

Whether or not limiting, Chow discloses claim [1pre]. Ex.1022 ¶¶ 138-142. Chow discloses claim [1pre]'s **method of effecting on a preexisting input file a computer-executable process comprised of a plurality of subtasks**. *Id.* ¶ 139. For example, Chow claims a "**method**" of "independently operating each one of multiple task execution units to sequentially and independently self-allocate and execute sub-tasks of the aggregate task on-demand." Ex.1003 (10:10-13); Ex.1022 ¶ 140. Chow further discloses this method, noting that "the method sequence 400 is used to perform an aggregate database processing task, by breaking the task down into smaller units, and executing the units with multiple concurrently operating task execution units." Ex.1003 (5:1-5); Ex.1022 ¶ 140.

Reiffin also discloses claim [1pre]. Ex.1022 ¶ 141. For example, Reiffin discloses that "a large compute-intensive task may be partitioned into subtasks which are then distributed among a plurality of personal computer or workstations for parallel execution of the subtasks." Ex.1004 (1:9-12); Ex.1022 ¶ 141.

For these reasons, the combination of Chow and Reiffin discloses every limitation **comprising** claim [1pre]'s **method of effecting on a preexisting input file a computer-executable process comprised of a plurality of subtasks**. Ex.1022 ¶ 142.

Thus, Chow and Reiffin both disclose or suggest claim [1pre]. Ex.1022 ¶ 142.

b. [1a] (a) automatically determining file allocation and logically subdividing records of said input file into a plurality of partitions;

This is a conventional step in parallel processing. Ex.1022 ¶ 143.

Chow discloses claim [1a]. Ex.1022 ¶ 144. For example, Chow discloses that "[t]ypically, workload is apportioned by somehow logically dividing the data to be processed. For example, a block of data might be divided evenly into a number of parts equal to the number of available threads, so that each thread can independently process a separate portion of the data. This is called "static" or "a priori" partitioning." Ex.1003 (1:38-42); Ex.1022 ¶ 144.

Chow further discloses:

> A command is received to perform a machine-executed task divisible into multiple sub-tasks, i.e., an "aggregate task". Then, multiple task execution units are independently operated on-demand to sequentially self-allocate and execute sub-tasks of the aggregate task. Each sub-task is sized according to a specific 'granularity', used to divide the aggregate task into sub-tasks. The granularity may be predetermined for all task execution units or repeatedly dynamically determined by each task execution unit at the time of self-allocation. Preferably, the granularity is not too large as to cause potentially uneven sub-task allocation, and not so small as to incur excessive overhead in allocating sub-tasks. Operating "on-demand", each task execution unit sequentially allocates and executes one sub-task at a time, then proceeding to allocate the next unallocated [piece] of work to itself according to the appropriate granularity.

Ex.1003 (1:66-2:15).

As a result, Chow discloses determining file allocation and logically subdividing records of said input file into a plurality of partitions. Ex.1022 ¶ 144.

Reiffin also discloses claim [1a]. Ex.1022 ¶ . For example, Reiffin discloses that "[t]his invention relates to computer networks, and more particularly, to a distributed parallel processing network system wherein a large compute-intensive task may be partitioned into subtasks." Ex.1004 (1:8-10). Reiffin further discloses: "If a computer has a task which is both large and compute-intensive, the task is

partitioned into a plurality of smaller subtasks to be executed simultaneously in parallel by the other computers of the network. Each of these subtasks is assigned its respective identifier" (Ex.1004 (2:19-23)) and "[t]he workstations or personal computer originally given such a task may partition the data and program code into a plurality of subtasks. A plurality of subtask identifiers corresponding to the partitioned subtasks are stored in a network directory of a disk drive of one of the computers to constitute a pool or queue identifying remote subtasks to be executed" (Ex.1004 (4:64-5:3)). Reiffin, therefore, discloses determining file allocation and logically subdividing records of said input file into a plurality of partitions. Ex.1022 ¶ 146.

Thus, Chow and Reiffin both disclose or suggest claim 1a. Ex.1022 ¶ 147.

c. [1b] (b) distributing descriptions of all of said partitions to each of a plurality of subtask processors

Chow teaches or suggests claim [1b]. Ex.1022 ¶ 149. Chow discloses that partitions are accessible by all subtask processors. For example, Chow discloses that "[t]he common partition is accessible by all task execution units 101-104." Ex.1003 (9:37-38). As such, each of the task execution units have access to information about the partition." Ex.1022 ¶ 149 . Chow further discloses: "[a] command is received to perform a machine-executed task divisible into multiple sub-tasks, i.e., an 'aggregate task'. Then, multiple task execution units are

independently operated on-demand to sequentially self-allocate and execute sub-tasks of the aggregate task. Ex.1003 (1:66-2:15).

Further, Reiffin teaches or suggests claim [1b]. Ex.1022 ¶ 150. Reiffin discloses that "[i]f a computer has a task which is both large and compute-intensive, the task is partitioned into a plurality of smaller subtasks to be executed simultaneously in parallel by the other computers of the network. Each of these subtasks is assigned its respective identifier. One of the network disk drives has a directory constituting a pool for storing all of the subtask identifiers, preferably in the form of a queue." Ex.1004 (2:19-25). Reiffin therefore discloses providing descriptions of the partitions to each of the subtask processors. Ex.1022 ¶ 150.

Thus, Chow and Reiffin both disclose or suggest claim [1b]. Ex.1022 ¶ 151.

d. [1c] (c) simultaneously executing at least a respective one of the subtasks of the computer-executable process in each of at least some of said processors on a respective one of the partitions with each subtask reading and processing the respective partition so as to process the respective partition and produce respective subtask output and;

This is not only a conventional step in parallel processing – it is the very essence of parallel processing. Ex.1022 ¶ 152.

Chow discloses that "[g]enerally, the method sequence 400 is used to perform an aggregate database processing task, by breaking the task down into smaller units, and executing the units with multiple concurrently operating task

execution units." Ex.1003 (5:1-5). A POSITA would understand Chow's "units" to be the claimed "partitions." Ex.1022 ¶ 153. Chow describes the concurrent execution of subtasks to produce a resulting output. Ex.1022 ¶ 153; Ex.1003 (7:46-48).

Reiffin discloses that "a large compute-intensive task may be partitioned into subtasks which are then distributed among a plurality of personal computer or workstations for parallel execution of the subtasks." Ex.1004 (1:9-12). Reiffin further discloses that "[e]ach of the other computers of the network selects a subtask identifier from the pool, copies the corresponding subtask program from the originating computer, executes the selected subtask in the background while concurrently executing its own local task in the foreground, returns the computed results of the subtask computation to the originating computer, and then selects another subtask from the pool to repeat the cycle." Ex.1004 (2:25-33). Reiffin discloses the parallel execution of subtasks from a partition with each subtask processor creating results. Ex.1022 ¶ 154.

Thus, Chow and Reiffin both disclose or suggest claim [1c]. Ex.1022 ¶ 155.

e. [1d] (d) thereafter repeating step (c) in at least some of the subtask processors each with another unprocessed partition on a first-come/first-served basis; and

This is a conventional step in parallel processing. Ex.1022 ¶ 156. Chow discloses that "[a]fter step 404, step 408 concurrently begins processing the

aggregate task by concurrently starting each of the task execution units 101-104 on different sub-tasks. When a task execution unit finishes its sub-task, it embarks on another sub-task, if any remain. Thus, the multiprocessing system 100 functions like a 'straw model', with several people drinking water from a common glass through individual straws." Ex. 1003 (5:13-21). Chow's disclosed straw method discloses that when a subtask processor completes a subtask, it selects a new subtask for processing, and this "straw model" is performed on a first-come/first-served basis. Ex.1022 ¶ 157.

Reiffin discloses that once a subtask processor completes a task, it returns the result and then selects another subtask to process: "[i]f a computer has a task which is both large and compute-intensive, the task is partitioned into a plurality of smaller subtasks to be executed simultaneously in parallel by the other computers of the network. Each of these subtasks is assigned its respective identifier. One of the network disk drives has a directory constituting a pool for storing all of the subtask identifiers, preferably in the form of a queue." Ex. 1004 (2:19-25). Reiffin's "queue" is a first-come-first-served line. Ex.1022 ¶ 158.

Thus, Chow and Reiffin both disclose or suggest claim [1d]. Ex.1022 ¶ 159.

f. [1e] (e) generating at least one output combining all of the subtask outputs and reflecting the processing of all of said subtasks.

This is a conventional step in parallel processing. Ex.1022 ¶ 160. Chow discloses that "[i]f step 510 finds that further processing still remains, the intermediate results are again pooled in step 506, and step 508 repeated. Steps 506 and 508 are repeated until processing of the data structure is completed. At this point, task 511 pools the results for final output, then, the routine 500 ends in step 512." Ex.1003 (7:66-8:4). Chow therefore discloses that the results are pooled into a final output, reflecting the processing of all the subtasks. Ex.1022 ¶ 161.

Reiffin similarly discloses that "The computed results from the execution of each subtask are placed into files which are stored in a network directory which may be accessed by the workstations which originally partitioned and distributed the compute-intensive task." Ex.1004 (6:9-13). Reiffin then also discloses generating output results, reflecting the processing of the subtasks, and storing them in a central location. Ex.1022 ¶ 163.

Thus, Chow and Reiffin both disclose or suggest claim [1e]. Ex.1022 ¶ 164.

Therefore, claim 1 was obvious over Chow and Reiffin separately and over their combination. Ex.1022 ¶ 165.

2. Claim 2: The method defined in claim 1 wherein the automatic determination of file allocation and logical subdivision of records of said input file into said plurality of partitions in step (a) and the distribution of the description of all of said partitions in step (b) is carried out with at least one further processor in addition to the subtask processors.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 166. The additional limitation of claim 2 is a conventional step in parallel processing. Ex.1022 ¶¶ 167.

Chow discloses that "[t]he present invention relates to multiprocessing computer database systems. More particularly, the invention concerns a method, apparatus, and article of manufacture for performing an aggregate database processing task with multiple concurrently operating task execution units, using incremental and on-demand sub-task allocation." Ex.1003 (1:10-15). Therefore, Chow discloses that the subtasks are processed by separate processors. In addition, Chow discloses a separate processor for partitioning: "Typically, workload is apportioned by somehow logically dividing the data to be processed. For example, a block of data might be divided evenly into a number of parts equal to the number of available threads, so that each thread can independently process a separate portion of the data. This is called 'static' or 'a priori' partitioning." Ex.1003 (1:38-42) Chow therefore discloses execution units, *i.e.*, subtask processors, that are independent from the computer that creates the partitions. Ex.1022 ¶ 168.

Reiffin discloses that a computer, different from the subtask processors, partitions the task into subtasks to be executed by "other computers on the network": "If a computer has a task which is both large and compute-intensive, the task is partitioned into a plurality of smaller subtasks to be executed simultaneously in parallel by the other computers of the network." Ex.1004 (2:19-22); Ex.1022 ¶ 169. A POSITA would have understood that all the computers involved in the processing of tasks and subtasks include processors.

Thus, Chow and Reiffin both disclose or suggest claim 2. Ex.1022 ¶ 170. Therefore, claim 2 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶ 171.

3. Claim 3: The method defined in claim 1, further comprising the step of merging said subtask outputs in step (e).

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 172. The additional limitation of claim 3 is a conventional step in parallel processing. Ex.1022 ¶ 173.

As stated with regard to claim [1e], Chow discloses that the results are pooled into a final output, reflecting the processing of all the subtasks. *See infra* claims [1e]; Ex.1003 (7:66-8:4); Ex.1022 ¶ 174.

And as further noted regarding claim [1e], Reiffin discloses generating output results, reflecting the processing of the subtasks, and storing them in a central location. *See infra* claims [1e]; Ex.1004 (6:9-13); Ex.1022 ¶ 175.

Thus, Chow and Reiffin both disclose or suggest claim 3. Ex.1022 ¶ 176. Claim 3 was therefore obvious over Chow and Reiffin separately or in combination. *Id.*

4. Claim 4: The method defined in claim 1 wherein the output in step (e) is a succession of outputs from said subtasks in a one-to-one correspondence with said records of said input file.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 177.

As discussed with regard to [1e], Chow discloses that the output, reflecting the processing of all the subtasks, is a succession of outputs from subtasks that relate directly to the partitions of the input file. *See infra* claims [1e]; Ex.1003 (7:66-8:4); Ex.1022 ¶ 179.

And as further noted regarding claim [1e], Reiffin discloses generating output results, reflecting the processing of the subtasks, and storing them in a central location where they relate directly to portions of the input file. *See infra* claims [1e]; Ex.1004 (6:9-13); Ex.1022 ¶ 180.

Thus, Chow and Reiffin both disclose or suggest claim 4 and therefore claim 4 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 181-182.

5. Claim 5: The method defined in claim 1 wherein the output in step (e) is an accumulation of output records from said subtasks in an arbitrary order.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 183.

As discussed with regard to [1e], Chow discloses that the output, reflecting the processing of all the subtasks, are accumulated. *See infra* claims [1e]; Ex.1003 (1:43-57; *see also* 9:6-21); Ex.1022 ¶ 185.

And as further noted regarding claim [1e], Reiffin discloses generating output results, reflecting the processing of the subtasks, and storing, *i.e.*, accumulating, them in a central location where they relate directly to portions of the input file. *See infra* claims [1e]; Ex.1004 (6:9-13); Ex.1022 ¶ 187.

Neither the Chow nor the Reiffin reference disclose a preference for whether the accumulation of the output is ordered or arbitrary. Ex.1022 ¶ 188. As a result, a POSITA would understand that the Chow and Reiffin references could accumulate the results in an ordered or arbitrary way. *Id*.

Thus, Chow and Reiffin both disclose or suggest claim 5 and therefore claim 5 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 189-190.

6. Claim 6: The method defined in claim 1 wherein said input file resides on a storage area network and is derived therefrom.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶¶ 191.

Chow discloses a multiprocessing system that includes a data storage unit and a data storage facility: "The multiprocessing system 100 includes multiple task execution units 101-104, each coupled to a data storage unit 106. . . . The data storage unit 106 comprises one or more devices for electronically storing data, such as an integrated circuit memory, magnetic hard drive, optical disk drive, magnetic or optical tape, etc. As described below, the task execution units 101-104 may advantageously be used for transferring data from the data storage unit 106 to another data storage facility, as well as performing optional manipulation of the data in transit. … Nonetheless, depending upon the needs of the application, the memory 108 may comprise any device for electronically storing data, such as an integrated circuit memory, magnetic hard drive, optical disk drive, magnetic or optical tape, etc." Ex.1003 (3:23-49). As storage area networks were common in

2002, a POSITA would have understood a storage area network to be among Chow's "any device for electronically storing data. Ex.1022 ¶ 193.

Reiffin also discloses uses of storage-area networks relating to the input file and subtasks: "If a computer has a task which is both large and compute-intensive, the task is partitioned into a plurality of smaller subtasks to be executed simultaneously in parallel by the other computers of the network. Each of these subtasks is assigned its respective identifier. One of the network disk drives has a directory constituting a pool for storing all of the subtask identifiers, preferably in the form of a queue." Ex.1004 (2:19-25); Ex.1022 ¶ 194.

Thus, Chow and Reiffin both disclose or suggest claim 6 and therefore claim 6 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 195-196.

7. Claim 7: The method defined in claim 1 wherein said input file resides on a network-attached storage and is derived therefrom.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 197.

As discussed with regard to claim 6, Chow discloses a network-attached storage for storing the input file. As network-attached storage was common in 2002, a POSITA would have understood network-attached storage to be among

Chow's "any device for electronically storing data. Ex.1022 ¶ 199. *See infra* claim 6; Ex.1022 ¶ 193.

Thus the combination of Chow and Reiffin discloses or suggests claim 7 and therefore claim 7 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 195-196.

8. Claim 8: The method defined in claim 1 wherein said computer-executable process is a sort process.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 202.

Because of the large size of datasets requiring sorting, one of the earliest applications of parallel processing was to sorting. A very old method of sorting known as "merge sort" is particularly amenable to parallel processing. Ex.1022 ¶¶ 203-204.

Chow discloses using its disclosed invention in the context of a sort process: "One mechanism to pool and redistribute data during multi-level processing is the partition sort technique, referred to as "p-sort". Broadly, the p-sort approach divides the data into sorted partitions according to a predetermined criteria, then assigns each partition to a task execution unit." Ex.1003 (8:9-13); Ex.1022 ¶ 205.

Reiffin explains that the disclosed invention can be applied to any of a variety of "large compute-intensive applications which frequently arise in science,

engineering, financial analysis, image processing and other fields." Ex.1004 (1:52-54). A POSITA would understand Reiffin's expansive potential use in any of a variety of applications would extend to include a sort process. Ex.1022 ¶ 206.

Thus, Chow and Reiffin both disclose or suggest claim 8 and therefore claim 8 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 207-208.

9. Claim 9: The method defined in claim 1 wherein said computer-executable process is a statistical analysis process.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 209.

Statistical analysis was one of the early applications of parallel processing. Ex.1022 ¶¶ 210.

Chow discloses that "By any name, engineers and scientists are constantly seeking new ways to minimize their applications task completion time." EX1003, 1:23-25. A POSITA would recognize that statistical analysis is an "application" of interest to engineers and scientists. Ex.1022 ¶ 211.

Reiffin explains that the disclosed invention can be applied to any of a variety of "large compute-intensive applications which frequently arise in science, engineering, financial analysis, image processing and other fields." Ex.1004 (1:52-54). A POSITA would understand Reiffin's expansive potential use in any of a

variety of applications would extend to include a statistical analysis process. Ex.1022 ¶ 212.

Thus, Chow and Reiffin both disclose or suggest claim 9 and therefore claim 9 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 213-214.

10. Claim 10: The method defined in claim 1 wherein said computer-executable process is a report-creating process.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 215.

Chow discloses that "By any name, engineers and scientists are constantly seeking new ways to minimize their applications task completion time." EX1003, 1:23-25. A POSITA would recognize that report generation is an "application" of interest to engineers and scientists. Ex.1022 ¶ 217.

Reiffin explains that the disclosed invention can be applied to any of a variety of "large compute-intensive applications which frequently arise in science, engineering, financial analysis, image processing and other fields." Ex.1004 (1:52-54). A POSITA would understand Reiffin's expansive potential use in any of a variety of applications would extend to include a report-creating process. Ex.1022 ¶ 218.

Thus, Chow and Reiffin together disclose or suggest claim 10 and therefore claim 10 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 219-220.

11. Claim 11: The method defined in claim 1 wherein said computer-executable process includes a database query.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 221.

It was well-known to use parallel processing for database operations. Ex.1022 ¶ 222. For example, Hu, discussed above in ¶¶ 59-64, was completely devoted to that topic in 1989. *Id.* Hu also cites 18 prior papers on parallel database processing. *Id.*

Chow is devoted to multiprocessor database systems:

> The present invention relates to multiprocessing computer database systems. More particularly, the invention concerns a method, apparatus, and article of manufacture for performing an aggregate database processing task with multiple concurrently operating task execution units, using incremental and on-demand sub-task allocation.

EX1003, 1:10-16; Ex.1022 ¶ 223.

Chow discloses use of a "Table Scan" operation that can be used to perform database query processing steps: "The "Table Scan" operation moves rows of tables from one storage location to another. Table Scan may be used, for example,

to retrieve table data from nonvolatile storage to RAM. This retrieval provides a process or "thread" with access to the data, so that the process can perform various operations on the data. These operations, for example, may involve sort, group, join, and other database query processing steps. Table Scan may be performed with granularity of one row, multiple rows, all rows on one page, or another appropriate unit of the underlying table data." Ex.1003 (6:49-59); Ex.1022 ¶ 224.

Reiffin explains that the disclosed invention can be applied to any of a variety of "large compute-intensive applications which frequently arise in science, engineering, financial analysis, image processing and other fields." Ex.1004 (1:52-54). A POSITA would understand Reiffin's expansive potential use in any of a variety of applications would extend to include a database query. Ex.1022 ¶ 225.

Thus, Chow and Reiffin both disclose or suggest claim 11 and therefore claim 11 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 226-227.

12. Claim 12: The method defined in claim 2 wherein said one processor is part of a mainframe computer and the other processors are processors of at least one other computer.

Chow and Reiffin disclose or suggest **the method defined in claim 2**. Ex.1022 ¶ 228.

The earliest parallel processing computers were mainframes. Ex.1022 ¶ 229.

Chow explains that the disclosed invention is intended to reduce task complete time in multiprocessing environments, which it describes as "computer systems where multiple computers, digital data processing units, threads, or other processes perform concurrent tasks." Ex.1003 (1:27-29); Ex.1022 ¶ 230. Chow further explains that "[e]ach task execution unit may also comprise a hardware device such as a computer, microprocessor, programmed memory, discrete circuitry, application-specific integrated circuitry, programmable circuit components, logic circuitry, a combination of the foregoing, or another suitable digital data processing machine." Ex.1003 (3:58-63). Chow therefore discloses that a multi-processing system can include multiple computers, with no limitation on their potential architecture. Ex.1022 ¶ 230. Therefore, a POSITA would understand that Chow could be implemented in an environment where the multiple computers include a mainframe and other computers. *Id.*

An objective of Reiffin is to create a network of a plurality of computers for parallel processing:

> It is a further object of the invention to provide a system wherein a network of workstations or personal computers may operate in parallel to form a powerful multicomputer system for parallel processing of large time-consuming compute-intensive applications in a fraction of the time that it would take any of the individual computers to do the job alone.

EX1004, 2:1-7

Reiffin thus discloses a multi-computer architecture. A POSITA would realize that a mainframe, being even more capable than a workstation, cud be employed in place of one of the workstations. Ex.1022 ¶ 232.

Thus, Chow and Reiffin together disclose or suggest claim 12 and therefore claim 12 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 233-234.

13. Claim 13: The method defined in claim 1 wherein said plurality of processors are all parts of a single multiprocessor.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶¶ 235.

Parallel processing on a single multiprocessor was disclosed at least by 1960 in Yarbrough (Ex.1009). It is also disclosed in Hurson (Ex.1013) and Koniges (Ex.1019). Ex.1022 ¶ 236.

Fig. 1 of Chow discloses an architecture with multiple task execution units sharing a common memory and a common data storage unit:

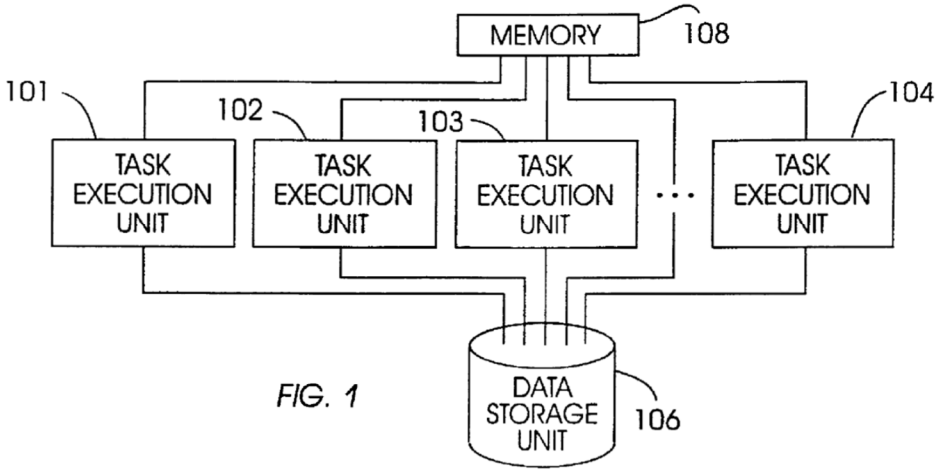


FIG. 1

A POSITA would recognize that the task execution units could all be processors of a single computer. In fact, such an organization would be natural in view of the shared memory and data storage unit. Ex.1022 ¶ 238.

Thus, Chow and/or Chow in combination with Reiffin disclose or suggest claim 13 and therefore claim 13 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 239-240.

14. Claim 14: The method defined in claim 1 wherein the automatic determination of file allocation and logical subdivision of records of said input file into said plurality of partitions in step (a) and the distribution of the descriptions of all of said partitions in step (b) is carried out with at least one processor, and said one processor and said plurality of processors are all parts of a single multiprocessor not including said one processor.

Chow and Reiffin disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 241.

As noted in Ex1022 ¶¶ 130-132, claim 14 is inoperable as written and incapable of enablement because it is logically contradictory. As such, it would be

impossible to locate any prior art over which an inoperable claim would be obvious. Ex.1022 ¶¶ 242.

The claim should have read, "The method defined in claim 1 wherein the automatic determination of file allocation and logical subdivision of records of said input file into said plurality of partitions in step (a) and the distribution of the descriptions of all of said partitions in step (b) is carried out with at least one processor, and said plurality of processors are all parts of a single multiprocessor not including said one processor." Ex.1022 ¶ 243. With that correction, the claim is obvious, and was disclosed in Yarbough (Ex.1009) in 1960, which distinguishes a "monitor processor" from the other processors. Ex.1022 ¶¶ 243. It is also disclosed in Koniges (Ex.1019), which refers to a "master processor." *Id.*

It therefore would have been obvious to a POSITA to use one of Chow's task execution units as a "monitor" or "master" processor. Ex.1022 ¶ 244.

Thus, Chow and Reiffin together disclose or suggest claim 14 and therefore claim 14 was obvious over Chow and Reiffin separately or in combination. Ex.1022 ¶¶ 245-246.

B. Ground 2: Claims 1-14 are obvious over Chow in view of Reiffin and further in view of Kurowski

Every limitation of claims 1-14 is disclosed or suggested by Chow (Ex.1003), Reiffin (Ex.1004), and Kurowski (Ex.1005). Ex.1022 ¶ 247. For all the

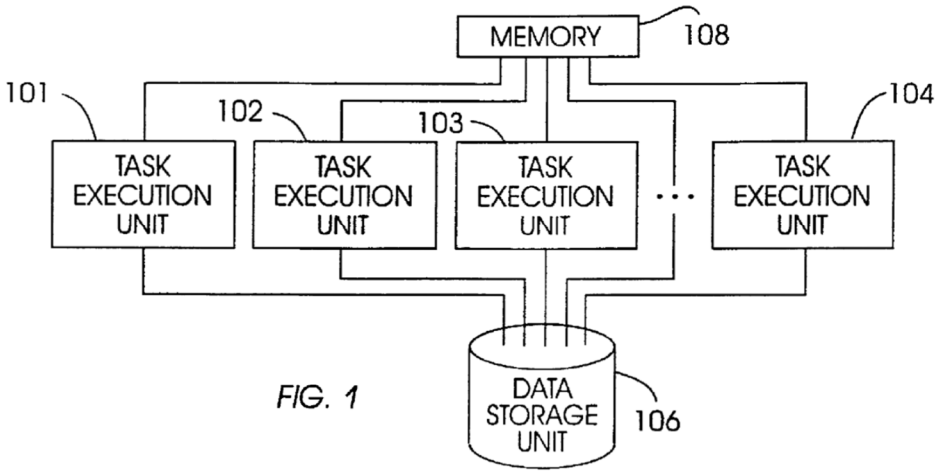
reasons previously stated it would have been simple to combine Chow's multi-processing system with Reiffin's teachings of a distributed parallel processing network resulting in the combination of known elements to obtain a predictable result, and thus such combination would be obvious to a POSITA. Ex.1022 ¶ 248.

In addition, Kurowski is also directed to the '582 Patent's goal of decomposing linear processes that currently use a single computer into equivalent parallel processes that can use any number of heterogeneous computers. *E.g.*, ("A distributed computing system achieves a highly distributed environment where very large computation intensive tasks are broken down into thousands of sub-tasks and then distributed to thousands of clients running on a variety of computers across the Internet.") Ex.1005 (Abstract). Ex.1022 ¶ 248.

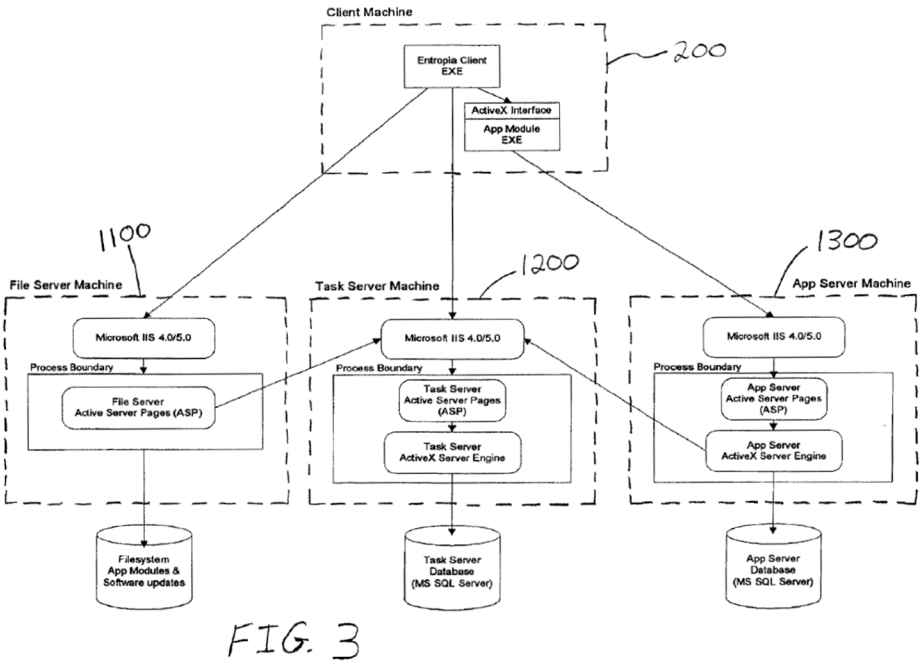
Both Chow and Reiffin were ready for improvement. Ex.1022 ¶ 249. Chow focuses on multiprocessor systems consisting of multiple processors. *Id.* Reiffin focuses on multiprocessor systems consisting of multiple separate computers connetce via a local area network. *Id.* Kurowski expands the concept to "thousands of clients running on a variety of computers across the Internet." *Id*.

It would have been routine to add the features of Reiffin to the system of Chow. Ex.1022 ¶ 250. It would have also been routine to add the features of Kurowski to Chow and Reiffin. *Id*.

In fact, the architectures of Chow and Kurowski are very similar in terms of task breakdown. Ex.1022 ¶ 250; *Cf.,* Fig. 1 of Chow and Fig. 3 of Kurowski:



EX1003, Fig. 1



EX1005, Fig, 3.

It was well-known as of the Priority Date how to interconnect heterogeneous computers over the Internet. Ex.1022 ¶ 251. This was one of the reasons the Internet was developed. *Id.*

A POSITA would have thus been motivated to combine Chow, Reiffin, and Kurowski with a reasonable expectation of success. Ex.1022 ¶ 252. Claims 1-14 are therefore obvious over Chow in view of Reiffin and further in view of Kurowski. *Id.*

1. Claim 1

a. [1pre] A method of effecting on a preexisting input file a computer-executable process comprised of a plurality of subtasks, the method comprising the steps of:

As previously noted in Ground 1, Chow and Reiffin disclose or suggest claim 1pre. Ex.1022 ¶ 253.

To the extent the Board disagrees that this is disclosed by the Chow/Reiffin combination, Kurowski discloses claim [1pre]'s **method of effecting on a preexisting input file a computer-executable process comprised of a plurality of subtasks**. *Id.* For example, Kurowski explains that "[t]he client 200 is one of the pieces in the larger picture of how the distributed computing system 100 described herein provides a highly distributed environment for large scale computing problems. One goal of all these different pieces is to enable the system to take on a very large computing problem, break it down into thousands of tasks and distribute

it to tens of thousands of computers across the internet so this large computation could be performed by tens of thousands of parallel sub-computations and the results brought back from all these different clients 200." Ex.1005 ([0192])

Chow, Reiffin, and Kurowski disclose every limitation **comprising** claim [1pre]'s **method of effecting on a preexisting input file a computer-executable process comprised of a plurality of subtasks**. Ex.1022 ¶ 256.

Thus, Chow, Reiffin, and Kurowski disclose or suggest claim [1pre]. Ex.1022 ¶ 256.

b. [1a] (a) automatically determining file allocation and logically subdividing records of said input file into a plurality of partitions;

As previously noted in Ground 1, Chow and Reiffin disclose or suggest claim 1a. Ex.1022 ¶ 257.

To the extent the Board disagrees that this limitation is disclosed by the Chow/Reiffin combination, Kurowski discloses claim [1a]. Ex.1022 ¶ 259. For example, Kurowski discloses that "[t]he client 200 is one of the pieces in the larger picture of how the distributed computing system 100 described herein provides a highly distributed environment for large scale computing problems. One goal of all these different pieces is to enable the system to take on a very large computing problem, break it down into thousands of tasks and distribute it to tens of thousands of computers across the internet so this large computation could be

performed by tens of thousands of parallel sub-computations and the results brought back from all these different clients 200." Ex.1005 ([0192]) Kurowski, therefore, discloses automatically identifying a large computing task and partitioning it.

Thus, Chow, Reiffin, and Kurowski disclose or suggest claim [1a]. Ex.1022 ¶ 260.

c. [1b] (b) distributing descriptions of all of said partitions to each of a plurality of subtask processors

As previously noted in Ground 1, Chow and Reiffin disclose or suggest claim 1c. Ex.1022 ¶ 262.

To the extent the Board disagrees that this limitation is disclosed by the Chow/Reiffin combination, Kurowski teaches or suggests claim [1b]. Ex.1022 ¶¶ 262-263. Kurowski discloses that "[s]pecifically, all machine, user, team, and task information, summaries of such information, or subsets of such information, may be made available through reports and interactive pages on an "integrated" web site." Ex.1005 ([0188]). As such, partition descriptions are distributed and made available to all of the processors. Ex.1022 ¶ 263.

Thus, Chow, Reiffin, and Kurowski disclose or suggest claim [1b]. Ex.1022 ¶ 264.

d. [1c] (c) simultaneously executing at least a respective one of the subtasks of the computer-executable process in each of at least some of said processors on a respective one of the partitions with each subtask reading and processing the respective partition so as to process the respective partition and produce respective subtask output and;

As previously noted in Ground 1, Chow and Reiffin disclose or suggest claim [1c]. Ex.1022 ¶ 265.

To the extent the Board disagrees that this limitation is disclosed by the Chow/Reiffin combination, Kurowski discloses that "[w]ith this system, one client 200 can work on one part of a calculation while a different client 200 works on another part of the same calculation. The distributed network of "servers" make the calculation all come together transparently." Ex.1005 ([0043]). Kurowski further explains that "[o]ne goal of all these different pieces is to enable the system to take on a very large computing problem, break it down into thousands of tasks and distribute it to tens of thousands of computers across the internet so this large computation could be performed by tens of thousands of parallel sub-computations and the results brought back from all these different clients 200." Ex.1005 ([0192]). Kurowski describes the concurrent execution of subtasks to produce a resulting output. Ex.1022 ¶ 267.

Thus, Chow, Reiffin, and Kurowski disclose or suggest claim [1c]. Ex.1022 ¶ 268.

e. [1d] (d) thereafter repeating step (c) in at least some of the subtask processors each with another unprocessed partition on a first-come/first-served basis; and

As previously noted in Ground 1, Chow and Reiffin disclose or suggest claim [1d]. Ex.1022 ¶ 269.

To the extent the Board disagrees that this limitation is disclosed by the Chow/Reiffin combination, Kurowski discloses that "[p]referably, the client 200 requests work rather than being given a task. In other words, the client 200 preferably requests or asks for work rather than receiving a start message from a server." Ex.1005 ([0069]). Kurowski further explains that "the client 200 preferably requests tasks from the Task Server 1200. At appropriate times in the client's execution, it may decide that its current task is finished and it needs to obtain a new task from the Task Server 1200. . . . In return, in step 222 the Task Server 1200 assigns a task to the client 200 based on information received from the client 200." Ex.1005 ([0154]). As a result, Kurowski discloses a system where subtasks are selected and processed on a first-come/first-served basis. Ex.1022 ¶¶ 271.

Thus, Chow, Reiffin, and Kurowski disclose or suggest claim [1d]. Ex.1022 ¶ 272.

f. [1e] (e) generating at least one output combining all of the subtask outputs and reflecting the processing of all of said subtasks.

As previously noted in Ground 1, Chow and Reiffin disclose or suggest claim [1e]. Ex.1022 ¶ 273.

To the extent the Board disagrees that this limitation is disclosed by the Chow/Reiffin combination, Kurowski discloses that "[w]ith this system, one client 200 can work on one part of a calculation while a different client 200 works on another part of the same calculation. The distributed network of "servers" make the calculation all come together transparently." Ex.1005 (6:9-13). Kurowski further explains that "[o]ne goal of all these different pieces is to enable the system to take on a very large computing problem, break it down into thousands of tasks and distribute it to tens of thousands of computers across the internet so this large computation could be performed by tens of thousands of parallel sub-computations and the results brought back from all these different clients 200." Ex.1005 ([0192]). Kurowski therefore discloses that the results are pooled into a final result. Ex.1022 ¶ 275.

Thus, Chow, Reiffin, and Kurowski disclose or suggest claim [1e]. Ex.1022 ¶ 276.

2. Claim 2: The method defined in claim 1 wherein the automatic determination of file allocation and logical subdivision of records of said input file into said plurality of partitions in step (a) and the distribution of the description of all of said partitions in step (b) is carried out with at least one further processor in addition to the subtask processors.

As previously noted in Ground 1, Chow and Reiffin disclose or suggest claim 2. Ex.1022 ¶ 277. Chow, Reiffin, and Kurowski disclose or suggest **the method defined in claim 1**. Ex.1022 ¶ 278.

To the extent the Board disagrees that this is limitation disclosed by the Chow/Reiffin combination, Kurowski discloses a partitioning and distribution system that is separate from the subtask processors: "The present invention advantageously addresses the needs above as well as other needs by providing a system for use in distributed computing. The system includes a task server, a file server, and an application server. The task server is configured to keep track of information associated with each of a multiplicity of client computers and to use the information to assign one or more tasks associated with a computing problem to each client computer. The file server is configured to provide application modules to the client computers for executing their assigned tasks. The application server is configured to provide input data for the application modules to the client computers and to receive output data of the application modules from the client computers." Ex.1005 ([0008]); Ex.1022 ¶ 281.

Thus, Chow, Reiffin, and Kurowski disclose or suggest claim 2. Ex.1022 ¶ 282.

3. Claims 3-14

Chow, Reiffin, and Kurowski disclose or suggest **the method defined in claim 1** and **the method defined in claim 2**. Ex.1022 ¶ 283. Further, as previously noted in Ground 1, Chow and Reiffin disclose or suggest claims 3-14. Ex.1022 ¶ 284. Therefore, Chow, Reiffen, and Kurowski disclose or suggest claims 3-14. *Id.* ¶ 285.

XI. THE BOARD SHOULD INSTITUTE *INTER PARTES* REVIEW

A. The *Fintiv* factors either favor institution or are neutral

Institution should not be denied under *Fintiv* because all factors favor institution or, at the very least, are neutral. Factor 1 is neutral because neither party has requested a stay. *See, e.g.*, *Sand Revolution II, LLC v. Continental Intermodal Grp.-Trucking LLC*, IPR2019-01393, Paper 24 at 7 (P.T.A.B. June 16, 2020) (informative). Factor 2 weighs against discretionary denial because no trial date is scheduled in IV's suit against American or in IV's suit against Southwest. *Id.* at 8-10.

Factor 2 is neutral because, although no trial date is yet set in IV's case against American, the schedule jointly submitted to the district court would provide for a trial more than eighteen months after the filing date of this petition.

And although trial is set for July 27, 2026 in the case against Southwest, that date often changes. For example, the most recent time-to-trial data show that the median time to trial for jury trials in the Western District of Texas is 34.9 months. Table T-3—U.S. District Courts–Trials Statistical Tables For The Federal Judiciary (December 31, 2024), U.S. Courts, https://www.uscourts.gov/data-news/data-tables/2024/12/31/statistical-tables-federal-judiciary/t-3 (last visited March 28, 2025). The Board has instituted IPR based on other petitions with similar facts. Ericsson Inc. v. XR Commc'ns LLC, IPR2024-00613, Paper 9, 33-34 (Oct. 9, 2024).

Factor 3 favors institution because the parties have expended limited resources in the litigations and Petitioners acted expeditiously—*i.e.*, by filing the petition within four months of the original complaints and well before the start of the of the claim construction process, in the early stages of fact discovery, and before expert discovery. *Sotera Wireless, Inc. v. Masimo Corp.*, IPR2020-01019, Paper 12 at 16-17 (P.T.A.B. Dec. 1, 2020) (precedential).

Factor 4 favors institution because Petitioners challenge all claims of the '582 Patent, although IV asserts only a subset in the litigation. Exs.1007, 1008, 1024, and 1025. Petitioners also stipulate that, if the petition is instituted, Petitioners will not pursue in the District Court Litigation any ground raised or that could have been reasonably raised in an IPR. *Sotera Wireless,* Paper 12 at 18.

Factor 5 favors institution because institution will promote judicial efficiency and integrity by reducing the number of issues to be addressed in the parallel litigations against American and Southwest. *Sand Revolution*, Paper 24 at 12.

Factor 6 favors institution because Petitioners have set forth a strong case of unpatentability. *Sand Revolution*, Paper 24 at 13.

B. An analysis under 35 U.S.C. § 325(d) also favors institution

The Board may consider whether "the same or substantially the same prior art or arguments previously were presented to the Office" in deciding whether to institute *inter partes* review. *Becton, Dickinson & Co. v. B. Braun Melsungen AG*, IPR2017-01586, Paper 8 at 17 (P.T.A.B. Dec. 15, 2017) (citing 35 U.S.C. § 325(d)). The Board answers that threshold question by reviewing *Becton, Dickinson*'s "factors (a), (b), and (d)." *Advanced Bionics, LLC v. Med-El Elektromedizinische Geräte GMBH*, IPR2019-01469, Paper 6 at 10 (P.T.A.B. Feb. 13, 2020). But even if the Board answers the threshold question affirmatively, it still considers whether institution is appropriate because "factors (c), (e), and (f)" show "a material error by the Office." *Id.*

None of the references cited in this petition were before the Examiner during prosecution. And Chow, Reiffin, and Kurowski do not "disclose[] substantially the same information" that the prior art the Examiner relied on does "in relevant part."

Advanced Bionics, Paper 6 at 15. During prosecution, Applicant argued that "[t]he primary difference between the instant invention and the processes disclosed in US 5,603,028 of Kitsuregawa and 5,357,632 of Pian is that these systems rely on a special control process that uses load information to distribute the load between processors that share the load. With the instant invention as defined in the claims there is no such special process. The prior art's load information is not created with the process of the instant invention. Instead, the load-sharing process take parts of the load on a first-come/first-served basis." As a result, Applicant's sole point of novelty over the prior art was that the distribution of subtasks was controlled rather than performed on a first-come/first-served basis.

Because Chow, Reiffin, and Kurowski all directly address the deficiency in the prior art considered by the Examiner—the *sole* purported point of novelty leading to issuance of the '582 Patent—they do not disclose substantially the same information as the cited prior art. *Maplebear Inc. v. Consumeron, LLC*, IPR2022-01357, Paper 9 at 14 (P.T.A.B. Mar. 3, 2023) (finding that references not "sufficiently cumulative" of the prior art considered during examination "to warrant discretionary denial under § 325(d)" because it "provide[d] additional material teachings not disclosed" by the considered prior art). Thus, the *Becton, Dickinson* factors (a), (b), and (c) support that neither the Examiner nor the Board

considered the same or substantially the same prior art or arguments that were previously presented. *Advanced Bionics*, Paper 6 at 9-10 & n.10.

Factors (c), (e), and (f) also show a material error by the Office, further supporting institution. *Id.* Those factors support institution because the Examiner never considered Chow, Reiffin, or Kurowski. But as Petitioners show above (*see supra* Grounds 1-2), Chow, Reiffin, and Kurowski have many specific teachings that are material to showing patentability. *Advanced Bionics*, Paper 6 at 21. The combined references and Petitioners' arguments put forward a strong case that the '582 Patent's claims are obvious, thus "warrant[ing] reconsideration of the prior art or arguments." *Id.* at 9 n.10.

XII. CONCLUSION

Petitioners respectfully request institution of IPR and cancellation of the Challenged Claims based on the grounds set forth and described above.

Dated: April 10, 2025

Respectfully submitted,

MCKOOL SMITH, P.C.
ATTN: John B. Campbell
303 Colorado Street
Suite 2100
Austin, TX 78701
Tel: (512) 692-8730
Fax: (512) 692-8744

By: */s/ John B. Campbell*
John B. Campbell
Reg. No. 54,665
Counsel for Petitioner

CERTIFICATE OF COMPLIANCE WITH WORD COUNT

Pursuant to 37 C.F.R. § 42.24(d), I certify that this petition complies with the type-volume limits of 37 C.F.R. § 42.24(a)(1)(i) because it contains 8,816 words, according to the word-processing system used to prepare this petition, excluding parts of this petition that are exempted by 37 C.F.R. § 42.24(a) (including the table of contents, a table of authorities, mandatory notices, a certificate of service or this certificate word count, appendix of exhibits, and claim listings).

Dated: April 10, 2025

MCKOOL SMITH, P.C.
ATTN: John B. Campbell
303 Colorado Street
Suite 2100
Austin, TX 78701
Tel: (512) 692-8730
Fax: (512) 692-8744

By: */s/ John B. Campbell*
John B. Campbell
Reg. No. 54,665

CERTIFICATE OF SERVICE

I hereby certify, pursuant to 37 C.F.R. Sections 42.6 and 42.106, that a complete copy of the attached **PETITION FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 7,257,582 B2**, including all exhibits **(Nos. 1001-1025)** and related documents, are being served by Federal Express on 10th day of April, 2025, the same day as the filing of the above-identified document in the United States Patent and Trademark Office/Patent Trial and Appeal Board, upon Patent Owner by serving the correspondence address of record with the USPTO as follows:

Brundidge & Stanger
2318 Mill Road, Suite 1020
Alexandria, VA 22314

A courtesy copy was also sent via electronic mail to the Patent Owner's litigation counsel listed below:

Jonathan K. Waldrop
jwaldrop@kasowitz.com
Darcy L. Jones
djones@kasowitz.com
Marcus A. Barber
mbarber@kasowitz.com
John W. Downing
jdowning@kasowitz.com
Heather S. Kim
hkim@kasowitz.com
ThucMinh Nguyen
tnguyen@kasowitz.com
Jonathan H. Hicks
Jhicks@kasowitz.com
KASOWITZ BENSON TORRES LLP
333 Twin Dolphin Drive, Suite 200
Redwood Shores, California 94065
Telephone: (650) 453-5170
Facsimile: (650) 453-5171

Dated: April 10, 2025

/s/ John B. Campbell
John B. Campbell
Reg. No. 54,665