UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

MICROSOFT CORPORATION,
Petitioner,

v.

PARTEC CLUSTER COMPETENCE CENTER GMBH,
Patent Owner.

Case IPR2025-00318
Patent 11,537,442 B2

PATENT OWNER'S PRELIMINARY RESPONSE UNDER 37 C.F.R. § 42.107 TO PETITION FOR *INTER PARTES* REVIEW

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

EXHIBIT LIST

Exhibit	Description
EX2001	Complaint for Patent Infringement, *PARTEC AG and BF EXAQC AG vs. MICROSOFT CORPORATION*, Civil Action No. 2:24-cv-433 (E.D. Tex. filed June 10, 2024)
EX2002	Second Amended Docket Control Order, Dkt. No. 48, *PARTEC AG and BF EXAQC AG vs. MICROSOFT CORPORATION*, Civil Action No. 2:24-cv-433 (E.D. Tex. filed April 9, 2025)
EX2003	United States District Courts – National Judicial Caseload Profile through December 31, 2024, https://www.uscourts.gov/sites/default/files/2025-02/fcms_na_distprofile1231.2024.pdf (accessed April 11, 2025)
EX2004	Defendant Microsoft's October 30, 2024 Invalidity Contentions, *PARTEC AG and BF EXAQC AG vs. MICROSOFT CORPORATION*, Civil Action No. 2:24-cv-433 (E.D. Tex.)
EX2005	Defendant Microsoft's March 6, 2025 Invalidity Contentions, *PARTEC AG and BF EXAQC AG vs. MICROSOFT CORPORATION*, Civil Action No. 2:24-cv-433 (E.D. Tex.)
EX2006	U.S. Patent Publication No. 2017/0262319
EX2007	Dynamic Process Management with Allocation-internal Co-Scheduling towards Interactive Supercomputing, Clauss et al., 1st COSH Workshop on Co-Scheduling of HPC Applications, 2016
EX2008	PDF comparison generated using PDF-XChange Editor of Petition Sections VII.C.1.c and VII.C.9.e against EX1003 Sections X.C.1.c and X.C.9.e
EX2009	Declaration of Dr. David Kaeli in Support of Patent Owner's Preliminary Response
EX2010	*Curriculum Vitae* for Dr. David Kaeli
EX2011	List of Cases for Dr. David Kaeli
EX2012	The DEEP-ER project: I/O and resiliency extensions for the Cluster-Booster architecture, Anke Kreuzer et al., 2018
EX2013	NVIDIA GeForce RTX 2080 Ti Specs | TechPowerUp GPU Database, https://www.techpowerup.com/gpu-specs/geforce-rtx-2080-ti.c3305, accessed April 11, 2025

EX2014	Efficient Data-parallel Distributed DNN Training for Big Dataset under Heterogeneous GPU Cluster, Shinyoung Ahn et al., 2024 IEEE International Conference on Big Data
EX2015	Practice and Experience in using Parallel and Scalable Machine Learning with Heterogenous Modular Supercomputing Architectures, Morris Riedel et al., 2021

TABLE OF CONTENTS

I. Introduction

Patent Owner ParTec AG[1] ("ParTec" or "Patent Owner") respectfully requests that the Board refuse to institute *inter partes* review of U.S. Patent No. 11,537,442 ("the '442 Patent") because Petitioner Microsoft Corporation ("Microsoft" or "Petitioner") has not shown a reasonable likelihood of prevailing on its challenge to Claims 1-10 of the '442 Patent (the "Challenged Claims"). The Petition grounds are based on a published patent application (US2013/0282787), invented by one of the '442 Patent co-inventors, that was the primary reference analyzed by the Examiner in the original prosecution. The Petition then relies on a secondary reference (Budenske) for limitations that Budenske itself expressly states that it cannot (and does not) teach. Even if that glaring omission could be overlooked, the Petition's obviousness theories rely on an unworkable combination that a POSITA would not have been motivated to attempt. The Budenske article from 1998 candidly acknowledges its many shortcomings including (1) its operating software (IOS) "has not been implemented" and represents "a major undertaking;" (2) the HC Kernel

[1] As noted in Patent Owner's Power of Attorney (Paper 3), the Patent Owner in this proceeding should be updated to ParTec AG, as reflected in USPTO assignment records. ParTec Cluster Competence Center GmbH (Reel 57967 Frame 304) changed its legal form and name to ParTec AG (Reel 67920 Frame 941).

requires off-line subtask mappings because of the longer execution times required; and (3) the HC Kernel implements subtasks using only "processors … of the same type." EX1005 at 389-390, 392. Petitioner apparently could not find any contemporary references to address the claimed limitations not taught by Lippert's own published application, and its attempts to fit a round peg in a square hole using the much older (and largely irrelevant) Budenske reference should be rejected.

Patent Owner is concurrently filing its bifurcated discretionary denial briefing as a separate paper pursuant to the Director's March 26, 2025 Memorandum. In addition to the arguments addressed below, the USPTO should exercise its discretion to deny institution for those reasons as well.

II. Background

Patent Owner ParTec is a leading provider of modular supercomputers and software that provide massive computing power for purposes such as artificial intelligence solutions. Its services include the design and development of high-performance computers (HPC), as well as consulting and support services in all areas of development, construction, and operation of these advanced systems. ParTec's modular supercomputing architecture represents a unique selling point for the company, and it has been hired to construct several of Europe's most powerful supercomputers.

A. State of the Art

A computing cluster generally includes a number of processing units and resources that can be organized to function as a single system. Computing clusters composed of multiple processors can be used to achieve improved data processing throughput and enhanced functionality relative to that available in a single computer. Initially, these clusters were homogenous and typically required a single type of general-purpose processor. EX2009 ¶¶28, 33.

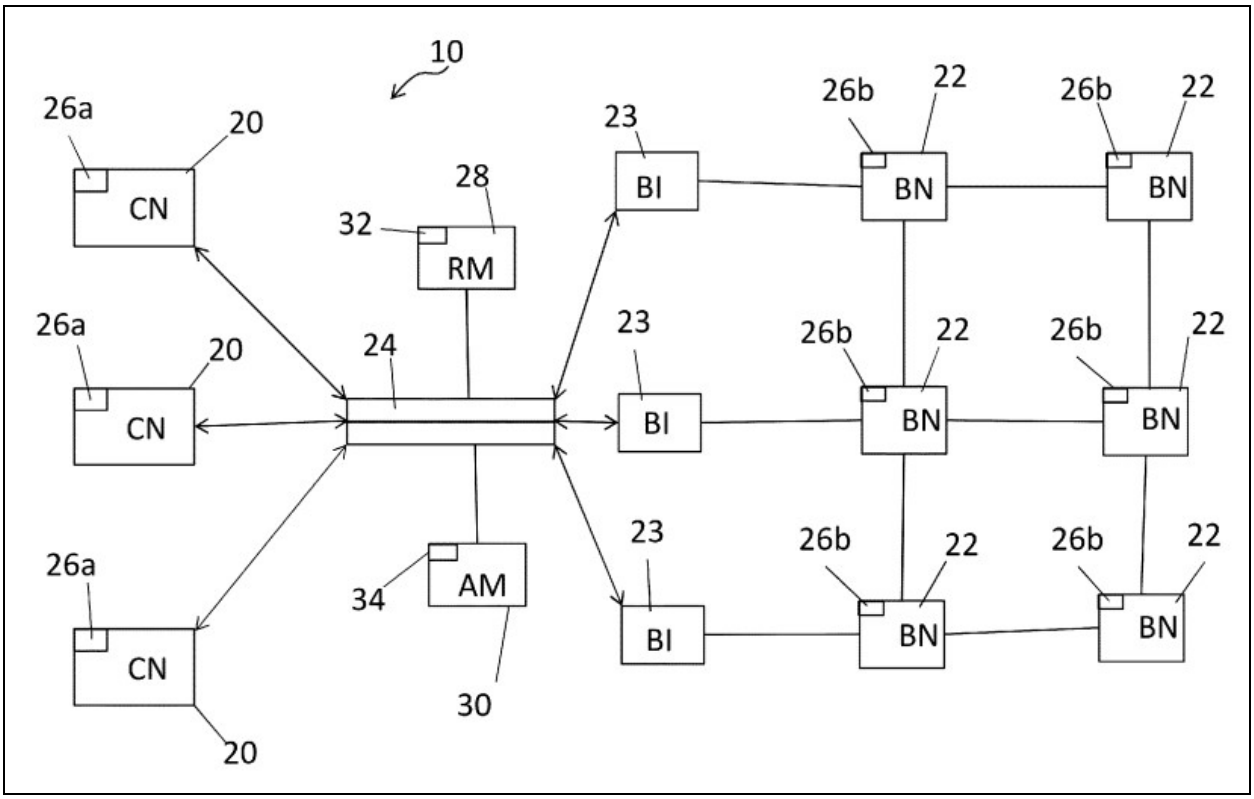
Different processor architectures provide different advantages and disadvantages. Traditional central processing units ("CPUs") typically offer flexibility and high single-thread clock speeds at the expense of high energy usage and limited parallelization capabilities. EX2009 ¶33. Other computing architectures have their own tradeoffs. For example, traditional graphical processing units ("GPUs") are more specialized, and may offer better parallel processing capabilities than CPUs, but may have slower clock speeds or less flexible processing architectures. *Id.* Certain computing tasks are more efficiently and/or more quickly handled by CPUs, while others are more efficiently and/or more quickly handled by specialized processors like GPUs. More specialized processors (such as GPUs) that offload processing from a CPU are sometimes referred to as "boosters" or "accelerators" because they provide the capability to perform specialized or more complex operations that would require significant CPU resources to perform. *Id.*

Heterogeneous systems enable the integration of multiple different processor architectures within a single system or cluster. EX2009 ¶¶28-30, 33. When dealing with heterogeneous systems, historically the question was to which type of processor a given subtask should be assigned because distribution of subtasks between different types of processors at the same time was not feasible. *See id.* ¶30. The '442 Patent inventors recognized that simply choosing one processor type over the other was insufficient to fully optimize operation of a cluster, and the far more useful (and difficult) question was how to efficiently utilize a combination of processor types (*e.g.*, a combination of computation nodes and booster nodes) dynamically over the course of several computing iterations. *See* EX1001 at 1:23-55, 4:64-5:24. It is this problem that the '442 Patent addresses. EX2009 ¶¶33-35.

B. The '442 Patent Invention

The '442 Patent is directed to a heterogeneous computing system arrangement capable of providing more efficient computation of tasks due—in part—to how it distributes sub-tasks across computing iterations. Heterogeneous computing systems incorporate different types of processors, often including general purpose processors along with more specialized "accelerator-type" processors. *See* EX1001 at 3:24-38.

An example embodiment is shown in Figure 1:



EX1001 Fig. 1.

As shown in Figure 1, the example system described in the ’442 Patent includes computation nodes 20 (“CN”) in communication with booster nodes 22 (“BN”) using communication infrastructure 24. *Id*. at 2:66-3:11. “The system 10 also includes a resource manager 28 shown connected to the communication infrastructure 24 and an application manager 30.” *Id*. at 3:19-21.

The ’442 Patent teaches how such heterogeneous computing systems can be adapted for parallel processing of large computation jobs. *See* EX1001 at 1:15-20. “A job to be computed by the system may comprise a number of tasks some of which or all may be repeated a number of times during the execution of the job.” *Id*. at

3:43-46. "The tasks may comprise a number of sub-tasks …[that] may be more or less suitable for processing by one or more of the computation nodes or one or more of the boosters. In particular, the scalability of the sub-task may indicate whether it is more appropriate for the sub-task to be processed by a computation node or a booster." *Id*. at 3:50-55. The '442 Patent gives the example of "a 'Monte-Carlo' based simulation where an effect is modelled using a random number, the calculations being repeated many times in succession." *Id.* at 3:44-49; EX2009 ¶34.

The '442 Patent teaches that efficiency can be improved by using a specific configuration capable of dynamically updating the distribution of sub-tasks between the CNs and BNs over the course of multiple computing iterations. EX1001 at 3:59-4:3. For example, even though a sub-task might appear suitable for a particular BN based on initial analysis or the results of earlier iterations, the sub-task might be better suited for a different distribution of computing resources on subsequent iterations. The '442 Patent teaches techniques to perform the monitoring and distribution based on "information learned about the processing of the sub-task and any need to call further sub-tasks during the processing." *See id.* at 4:48-5:9.

Independent claims 1 and 9 are reproduced below utilizing the same limitation numbering adopted by Petitioner. The last limitation of both Claim 1 and Claim 9 are similar and require providing an updated distribution of the sub-tasks among

computation nodes and booster nodes for processing a further computing iteration based on the claimed information:

> [1.1] A method of operating a heterogeneous computing system comprising a plurality of computation nodes and a plurality of booster nodes, at least one of the plurality of computation nodes and a plurality of booster nodes being arranged to compute a computation task, the computation task comprising a plurality of sub-tasks, the method comprising:
>
> [1.2] in a first computing iteration, assigning and processing the plurality of sub-tasks by at least a portion of the plurality of computation nodes and at least a portion of the plurality of booster nodes in a first distribution; and
>
> [1.3] generating, using information relating to the processing of the plurality of sub-tasks by at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes, a further distribution of the plurality of sub-tasks between the plurality of computation nodes and the plurality of booster nodes for processing thereby in a further computing iteration.

> [9.1] A heterogeneous computing system comprising:
>
> [9.2] a plurality of computation nodes and a plurality of booster nodes for computing one or more tasks comprising multiple sub-tasks;
>
> [9.3] a communication interface connecting the plurality of computation nodes with each other and the plurality of booster nodes;
>
> [9.4] a resource manager for assigning at least a portion of the plurality of booster nodes and at least a portion of the plurality of computation

nodes to each other for the computing of the one or more tasks in a first computing iteration; and

[9.5] an application manager configured to receive information from daemons operating in at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes to update a distribution of the multiple sub-tasks between the plurality of computation nodes and the plurality of booster nodes in a further computing iteration.

C. Petitioner's References

Petitioner asserts two grounds based on three references as shown in the table below. Pet. at 2.

Ground	References	Statute	Claims
Ground 1	Lippert (EX1004) and Budenske (EX1005)	35 U.S.C. § 103	1-10
Ground 2	Lippert, Budenske, and Kambatla (EX1006)	35 U.S.C. § 103	2-5, 8-10

1. Lippert [EX1004]

"Lippert" is a U.S. Patent Application (Pub. No. 2013/0282787) to Thomas Lippert titled "Computer Cluster Arrangement For Processing A Computation Task And Method For Operation Thereof." EX1004 at 1. Thomas Lippert is one of the inventors of the '442 Patent challenged in this proceeding, and the patent that issued from the 2013/0282787 patent application (10,142,156) is also assigned to ParTec. The Lippert patent application was addressed extensively during prosecution of the

'442 Patent, and its disclosure is incorporated by reference into the '442 Patent as "WO 2012/049247." *See, e.g.,* EX1002 at 156-166, 173-175; EX1001 at 1:23-24; Section II.D.

Lippert discloses a computer cluster arrangement including computation nodes ("CN") coupled to boosters ("B") using a communication infrastructure ("IN"). EX1004 at [0067]. A Resource Manager ("RM") can be used in Lippert to assign boosters to computation nodes. An embodiment of this arrangement can be seen in Lippert's Figure 2:

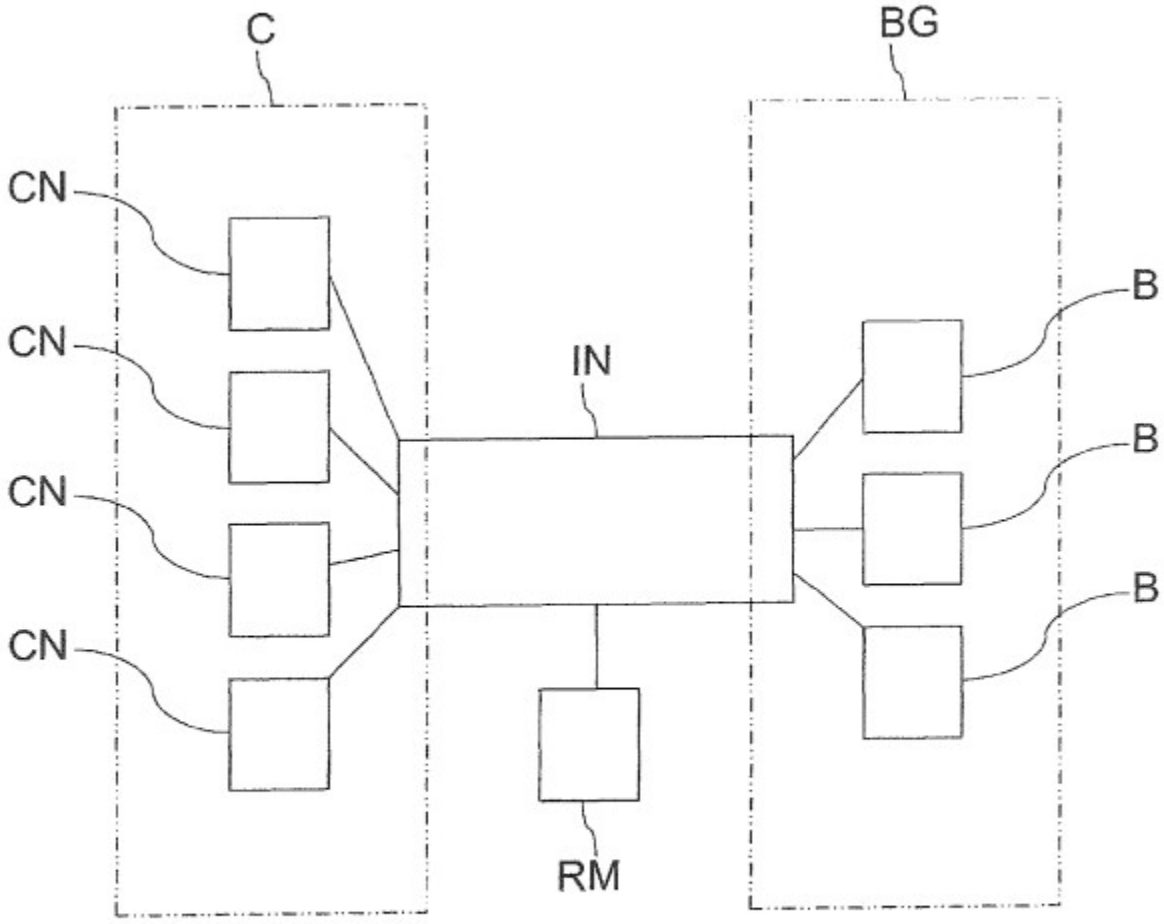


Fig.: 2

It is not surprising that Lippert and the ’442 Patent share some similarities given the ’442 Patent notes that it is a further development of Lippert. EX1001 at 1:23-28. Nor is it surprising that the ’442 Patent inventors continued to work to significantly improve on Thomas Lippert’s earlier work, leading to the claimed advances in the ’442 Patent. *See* EX1001 at 1:35-38. Petitioner’s use of Lippert as the primary reference in this proceeding (despite being the central focus during the

original prosecution) demonstrates just how groundbreaking the Lippert disclosure was and how deficient the other available prior art was in comparison to ParTec's own work. However, as the Patent Office concluded, Lippert's earlier patent does not teach or render obvious all of the improvements of the '442 Patent, as addressed in section II.D.

2. Budenske [EX1005]

"Budenske" is an article listing authors John R. Budenske, Ranga S. Ramanujan and Howard Jay Siegel titled "A Method for the On-Line Use of Off-Line Derived Remappings of Iterative Automatic Target Recognition Tasks onto a Particular Class of Heterogeneous Parallel Platforms." EX1005 at iv. Budenske identifies itself as being included in the October 1998 Edition of The Journal of Supercomputing. *Id.* at i-iii.

"Th[e] paper concentrates on the operation of the [heterogeneous computing ("HC")] Kernel" which it claims "differs from other real-time HC mapping techniques in that it allows on-line real-time use of off-line precomputed mappings." *Id.* at 388-389. The paper states that it "focuses on (1) the application and hardware platform characteristics that are needed to enable the use of the HC Kernel, (2) the techniques that comprise the HC Kernel, and (3) how the information needed for the HC Kernel to operate is collected." *Id.* The authors emphasize that the OS itself has

never been implemented, is a "major undertaking" and the paper relates only to "design concepts." *Id.* at 389.

Petitioner has relied on Budenske to address distributing sub-tasks between CNs and BNs in a further computing iteration after the initial iteration. *See, e.g.,* Pet. at 29-32. While Budenske may be directed to a "heterogeneous" system to the extent that it can include different types of processors, it expressly notes that its system was not capable of distributing tasks to a ***mix*** of processor types. EX1005 at 392 ("[I]f an implementation of a given subtask uses multiple processors, all processors will be of the same type."). Petitioner's suggestion that Budenske somehow suggests and would motivate a skilled artisan to dynamically assign sub-tasks among different types of processors on subsequent computing iterations is nothing more than attorney argument and requires a wholesale redesign of the Budenske disclosure. As explained in more detail below, distributing sub-tasks between processors of different types is difficult, and a POSITA at the applicable time would have understood Budenske to (at best) be determining which processor type to use for the full task—not dynamically distributing sub-tasks between disparate processor types.

3. Kambatla [EX1006]

"Kambatla" is a U.S. Patent Application (Pub. No. 2018/0074855) to Karthik Kambatla titled "Utilization-Aware Resource Scheduling in a Distributed Computing Cluster." EX1006 at 1. Kambatla relates to computing clusters, but does

not teach anything directed to heterogeneous computing using different types of processors. *See, e.g., id.* at [0002]. Kambatla never discusses the use of different processor types, distribution of tasks to different processor types, or anything else that might make it relevant to an actual heterogeneous computing architecture. Thus, Kambatla cannot remedy any of the deficiencies related to distribution of sub-tasks between disparate processor types.

D. ’442 Patent Prosecution History

As Petitioner concedes, the USPTO has already extensively evaluated its primary reference (Lippert) during prosecution of the ’442 Patent. *See* Pet. at 6-7. The Examiner initially rejected all pending ’442 Patent Claims over Lippert in an April 22, 2022 Office Action. EX1002 at 154-167. The applicant then amended the claims and traversed the rejection, noting Lippert failed to teach redistribution of sub-tasks based on information related to the processing of the plurality of sub-tasks across multiple computing iterations. EX1002 at 171-175. Thus, “Lippert fail[ed] to disclose the claimed combination ‘in a first computing iteration, assigning and processing the plurality of sub-tasks by at least a portion of the plurality of computation nodes and at least a portion of the plurality of booster nodes in a first distribution,’ and ‘generating, using information relating to the processing of the plurality of sub-tasks by at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes, a further distribution of the

plurality of sub-tasks between the plurality of computation nodes and the plurality of booster nodes for processing thereby in a further computing iteration,' as recited in amended claim 1 and as similarly recited in amended claim 9." *Id.* at 175. The Examiner agreed, allowing the amended claims over Lippert. *Id.* at 184. Petitioner is essentially seeking a mulligan of the Examiner's decision, using secondary references that fail to teach or suggest the aspects of the '442 claims which distinguished over the prior Lippert patent application.

III. Legal Standards

An IPR should not be instituted unless Petitioner has shown a likelihood of success on the invalidity grounds *presented in the petition*. *See In re Magnum Oil Tools Int'l, Ltd.*, 829 F.3d 1364, 1381 (Fed. Cir. 2016) ("[T]he Board must base its decision on arguments that were advanced by a party, and to which the opposing party was given a chance to respond.").

"In an IPR, the petitioner has the burden from the onset to show with particularity why the patent it challenges is unpatentable." *Harmonic Inc. v. Avid Tech.*, Inc., 815 F.3d 1356, 1363 (Fed. Cir. 2016) (petitions must identify "with particularity . . . the evidence that supports the grounds for the challenge to each claim"); 35 U.S.C. § 312(a)(3). This burden of persuasion never shifts to the patent owner. *See Dynamic Drinkware, LLC v. Nat'l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015).

"To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness." *In re Magnum Oil Tools Int'l*, 829 F.3d at 1380. The obviousness inquiry requires considering whether one of skill in the art "would have been motivated to combine the prior art to achieve the claimed invention." *In re NuVasive, Inc.*, 842 F.3d 1376, 1381 (Fed. Cir. 2016) (quoting *In re Warsaw Orthopedic, Inc.*, 832 F.3d 1327, 1333 (Fed. Cir. 2016)). "[T]he factual inquiry whether to combine references must be thorough and searching…" *Id.*

IV. Level of Ordinary Skill in the Art

For purposes of this Preliminary Response, Patent Owner has applied Petitioner's recitation of the level of skill in the art, because even under Petitioner's proposed level of skill, Petitioner has failed to demonstrate a reasonable likelihood of success. *See* Pet. at 8. Patent Owner reserves the right to propose its own definition of a person of ordinary skill in the art ("POSITA") in the future if necessary.

V. Claim Construction

Claims are construed according to *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (en banc). *See* 37 C.F.R. § 42.100(b). "Any prior claim construction determination concerning a term of the claim in a civil action…will be considered."

Id. For purposes of this Preliminary Response, Patent Owner does not believe any claim constructions are necessary.[2]

VI. Argument

The Petition should be denied for multiple independent reasons. First, both grounds fail to teach or render obvious limitations [1.3] and [9.5] regarding re-distribution of sub-tasks between CNs and BNs during a further computing iteration using information obtained during the processing of sub-tasks. Second, a POSITA would not have been motivated to combine Budenske with either Lippert or Kambatla.

A. Limitations [1.3] and [9.5]

1. Ground 1 fails to teach or render obvious the "further computing iteration" limitations

Independent Claim 1 requires:

> [1.3] generating, using information relating to the processing of the plurality of sub-tasks by at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes, a further distribution of the plurality of sub-tasks between the plurality of computation nodes and the plurality of booster nodes for processing thereby in a further computing iteration.

[2] Patent Owner reserves the right to propose specific constructions in the event that the Petition is instituted.

EX1001 Claim 1 at 5:26-44.

Similarly, independent Claim 9 requires:

> [9.5] an application manager configured to receive information from daemons operating in at least the portion of the plurality of computation nodes and at least the portion of the plurality of booster nodes to update a distribution of the multiple sub-tasks between the plurality of computation nodes and the plurality of booster nodes in a further computing iteration.

EX1001 Claim 9 at 6:37-43.

As shown above, both limitations relate to distribution of sub-tasks between CNs and BNs in a further computing iteration. For brevity, this Preliminary Response refers to these collectively as the "further computing iteration" limitations based on the shared last three words.

Petitioner does not contend that Lippert teaches any of limitations [1.3] or [9.5]. *See* Pet. at 29-34, 52 (citing only Budenske). Nor could it—as noted by the applicant during prosecution (and agreed to by the examiner), Lippert fails to teach generating further distributions of sub-tasks between the computation nodes and booster nodes in further iterations by using information relating to the processing of the plurality of sub-tasks. EX1002 at 173-175, 184.

Instead, Petitioner is wholly reliant on Budenske for these limitations. However, Budenske fails to suggest and is incapable of distributing sub-tasks between different types of processors, much less of performing such a distribution

of sub-tasks among different processor types in a further computing iteration. In fact, Budenske expressly states that for its teachings to work "it is assumed that if an implementation of a given subtask uses multiple processors, **all processors will be of the same type**." EX1005 at 392 (emphasis added). By restricting execution to a single processor type, Budenske can avoid dealing with the differences in execution properties and speeds between processor types and instead treat "the expected execution time of a particular multiprocessor implementation of a subtask [as] independent of which fixed-size subset of the processors **of a given type** are assigned to execute the subtask." *Id.* (emphasis added); EX2009 ¶¶49, 56.

As Dr. Kaeli explains, in addition to the express statements from Budenske that it is limited to processors of a single type, a POSITA would also have understood the same restriction relative to Budenske from the discussion of its architecture. EX2009 ¶¶48-49. For example, Petitioner refers to the SHARC DSPs of Budenske as the accelerators/boosters. Pet. at 21. Budenske explains that, under its architecture, those DSPs will "physically share a DRAM." EX1005 at 392. Dr. Kaeli notes that in the context of Budenske, a POSITA would understand that those DSPs "all share the same working memory, and if five processors were needed instead of four for the next iteration (such as when the number of objects of interest in the last frame changed from four to five—*see* example on page 393), all DSP processors would have access to the same data set and could take over the extra processing load.

The same would not be true of different processor types (such as the RISC processors) which would not have direct access to that shared memory pool." EX2009 ¶49. Stated more simply, the Budenske architecture implements a shared memory that requires the processor types to be the same. The net effect is that a skilled artisan would conclude from Budenske that mixing processor types for particular sub-tasks would be difficult to implement and would require further innovations.

Budenske continues that same line of reasoning, emphasizing that processors of the same type will have symmetric and conflict-free intercommunication. EX1005 at 392. Budenske also emphasizes that expected execution times are independent of the assigned processor because it requires assignment to a single type of processor. *Id*. ("[T]he expected execution time of a particular multiprocessor implementation of a subtask is independent of which fixed-size subset of the processors **of a given type** are assigned to execute the subtask."). As Dr. Kaeli explains, all of this would indicate to a POSITA that Budenske could not assign a subtask to more than one type of processor. EX2009 ¶¶48-49.

While Budenske suggests that its disclosure addresses "how to assign resources (e.g., processors) to the subtasks" generally, the fact that the authors felt the need to expressly limit their discussion to using only one type of processor for any given sub-task speaks volumes of the difficulty of distributing sub-tasks between

disparate types of processors. *See* EX1005 at 391. This difficulty would have lead a POSITA to conclude that distributing subtasks between different types of processors was too difficult of a problem to have been solved without further innovations. EX2009 ¶¶28-29, 35-39.

Further confirming the difficulty of mapping subtasks to different processor types, Dr. Kaeli cites several publications that discuss the difficulty of mapping to multiple types of processors, including an article addressing the difficulty of mapping to multiple different types of GPUs resulting in what it refers to as a "straggler problem" that results in "lower GPU utilization", EX2009 ¶36 (citing EX2014 at 179-180). He also cites another article which notes that code that must utilize two different architectures (e.g., general purpose processor as well as a booster) "introduces significantly more complexity than was seen under [] other [] historical models, 'leading to a dissonance with these traditional HPC system workloads.'" EX2009 ¶37 (quoting EX2015 at 1-2). These publications further confirm that a POSITA would not have understood the Budenske disclosure to teach mapping of subtasks to different types of processors at the same time.

Thus, a POSITA would understand that Budenske could not (and did not teach) the limitations for which Petitioner relies on it. As a result, the Petition grounds cannot present a reasonable likelihood of invalidating any Challenged Claims.

2. Ground 2 fails to teach or render obvious the "further computing iteration" limitations

Petitioner's Ground 2 suffers the same deficiencies as Ground 1, and therefore also fails. Ground 2 relies on the same Lippert and Budenske references as Ground 1 (frequently citing back to the same Ground 1 analysis), but also adds Kambatla for certain limitations. Ground 2 does not address independent Claim 1 at all, and thus fails to remedy the Claim 1 shortcomings in Ground 1. For limitation [9.5] of independent Claim 9, Petitioner does not directly cite anything from Kambatla, but instead refers back to "Sections VII.C.1, VIII.B.1, VIII.B.4." Pet. at 72. As discussed below, none of those remedy the failures in Petitioner's Ground 1 analysis.

Petition Section VII.C.1 is Petitioner's Ground 1 analysis for Claim 1, and as discussed above that fails to teach or render obvious the "further computing iteration" limitations.

Petition Section VIII.B.1 is Petitioner's Ground 2 analysis of Claim 2, and Petition Section VIII.B.4 is Petitioner's Ground 2 analysis of Claim 5. Pet. at 62-65, 68-70. Both cite to Kambatla for support regarding certain limitations in dependent claims, but nothing cited in either section (or anywhere else) could teach or render obvious updating distributions of sub-tasks to CNs and BNs on further computing iterations. As an initial matter, Kambatla does not even address heterogeneous computing concepts, thus there is no disclosure addressing distribution of sub-tasks to different processor types in any context. EX2009 ¶¶50-52, 60. Petitioner

incorrectly characterizes Kambatla as relating to the "field of endeavor of executing computation tasks within a heterogeneous computing environment," however Petitioner cites nothing in Kambatla which demonstrates heterogeneity. *See* Pet. at 53.

For example, Petitioner cites Kambatla paragraphs [0002] and [0004] in support of its statement, but both paragraphs address only cluster computing concepts with nothing specifically related to heterogeneous computing or assigning tasks to different types of processors. Pet. at 53. Petitioner's expert defines cluster computing simply as "a collection of interconnected computing devices ("nodes") utilized as a single, unified computing resource." EX1003 ¶27. While a cluster can be implemented with multiple types of processors, there is no expectation that a cluster will necessarily include multiple types of processors. *See* EX2009 ¶¶28, 33. Nor is there an expectation that concepts related in general to cluster computing would be applicable to heterogeneous computing systems. As the above discussion of Budenske illustrates, using different types of processors to work on the same sub-tasks raises complex issues regarding system architecture, which most designers (like Budenske) would prefer to avoid. Even assuming for the sake of argument that some of Kambatla's techniques were applicable to heterogeneous computing, it certainly does not address distributing sub-tasks between different types of processors during further computing iterations. EX2009 ¶¶50-52, 60.

B. A POSITA would not have modified the references in the claimed manner

As discussed above, the Petition grounds fail to teach or render obvious at least limitations [1.3] and [9.5]. Furthermore, even if Budenske had taught limitations [1.3] and [9.5], a POSITA still would not have modified Lippert and Kambatla with the teachings of Budenske.

As discussed above, Budenske actually teaches away from distributing sub-tasks between processor types, noting that for its teachings to work all processors utilized for a given task must be of the same type. EX1005 at 392; Section VI.A.1. "'A reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant.'" *Allergan, Inc. v. Sandoz Inc.*, 796 F.3d 1293, 1305 (Fed. Cir. 2015) (quoting *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir. 1994)). Budenske explaining that for its solution to work it must be "assumed" that "if an implementation of a given subtask uses multiple processors, all processors will be of the same type" would discourage a POSITA dealing with distributing tasks to disparate types of processors from "following the path set out in the reference." EX1005 at 392; *Allergan*, 796 F.3d at 1305.

And even if the Board were to conclude that Budenske does not expressly teach away from assigning sub-tasks to different processor types, Budenske's

teachings would have strongly weighed against motivating a skilled artisan to develop a system according to the '442 Patent claims. That conclusion is further reinforced by subsequent publications addressing the Budenske reference, which confirm a POSITA would not have had a reasonable expectation of success of combining its dynamic parameter teachings with other references.

To demonstrate obviousness, Petitioner must show "that a skilled artisan would have had reason to combine the teaching of the prior art references to achieve the claimed invention, and that the skilled artisan would have had a reasonable expectation of success from doing so." *Redline Detection, LLC v. Star Envirotech, Inc.*, 811 F.3d 435, 449 (Fed. Cir. 2015) (internal quotation marks and citations omitted). One of the authors of Budenske (Howard Jay Siegel) noted in a subsequent publication that the purported methods of mapping tasks using "dynamic parameter values" in Budenske were not "ever evaluated in any way, as this was not the focus of Budenske et al." EX1007 at 79. Rather, the focus of Budenske was strictly on "[t]he design of the ATR Kernel and HC Kernel." *Id.* Thus, the authors of Budenske recognized and stated publicly that the "dynamic parameter values" and purported mappings that Petitioner relies on were not evaluated or expected to be usable, and a POSITA would have had no expectation of success in using them as taught by Budenske, much less in modifying a combination of Lippert and Budenske to use them in a manner that no reference of record actually taught was possible. The lack

of an expectation of success is further reinforced by references cited by Dr. Kaeli that emphasized the difficulty of mapping to processors of different types. *See* Section VI.A.1; EX2009 ¶¶28-29, 35-39.

The non-obviousness of the proposed combination is further reinforced by Kambatla. EX2009 ¶67. Kambatla does not even address the challenges of distributing sub-tasks in a more complicated heterogeneous architecture. But even in the homogeneous architecture contemplated by Kambatla, the reference noted that "[t]he amount of computing resources required to process a given task can be difficult to predict. It is inevitably difficult to accurately estimate the resource requirements of a job or its constituent tasks because: (i) resource usage of a task varies over time, and (ii) resource usage can vary across tasks of the same job based on the input they process." EX1006 at [0005]. Those challenges would be compounded even further in the context of a heterogeneous architecture, wherein resource usage of a task varies not just by time and input, but based on the varied capabilities of different processor types as well. EX2009 ¶67. Thus, even years after Budenske, Petitioner's other references continued to reinforce the difficulty of estimating resource usage and mapping tasks (let alone mapping sub-tasks across disparate processor types during subsequent computing iterations). This further reinforces that a POSITA would not have considered it to have been obvious to combine Lippert or Kambatla with the teachings of Budenske.

Even the significant gap in time between Budenske and Lippert/Kambatla demonstrates that there was no motivation to combine the references. Budenske identifies a purported publishing date of 1998, while Lippert claims a priority date in 2010 and Kambatla claims a priority date in 2016. If the proposed combination was as obvious, workable, and beneficial (without the use of hindsight) as Petitioner suggests, then a POSITA would have adapted the teachings of Budenske to Lippert well before the 2018 priority date of the '442 Patent. "The length of the intervening time between the publication dates of the prior art and the claimed invention can [] qualify as an objective indicator of nonobviousness." *Leo Pharm. Prods. v. Rea*, 726 F.3d 1346, 1359 (Fed. Cir. 2013). Correspondingly, the "intervening time between the prior art's teaching of the components and the eventual preparation of a successful composition speaks volumes to the nonobviousness of the [patent]." *Id.*; *see also* Merck & Co. v. Teva Pharm. USA, Inc., 395 F.3d 1364, 1376 (Fed. Cir. 2005) ("[T]he law presumes an idea would successfully have been brought to market sooner, in response to market forces, had the idea been obvious to persons skilled in the art.").

The effects of this gap in time are further amplified by the extreme difference in the state of the art when Budenske was published. As Dr. Kaeli notes, at the time Budenske was purportedly published a heterogeneous system was typically customized for a very specific task (automatic target recognition in the case of

Budenske). EX2009 ¶¶30-31, 48-49, 66. This meant a task or subtask would (at best) be assigned to the most appropriate processor type, rather than to a mix of processors. *Id.* That again conforms with Budenske's express teachings that a subtask would only be assigned to processors of a single type. EX1005 at 392.

VII. Conclusion

Patent Owner respectfully requests that the Board refuse to institute *inter partes* review for the reasons stated herein.

Dated: April 16, 2025

Respectfully submitted,

By: */Michael F Heim /*
Michael F. Heim (Reg. No. 32,702)
Attorney for Patent Owner
ParTec AG

CERTIFICATE OF SERVICE

The undersigned certifies that pursuant to 37 C.F.R. § 42.6(e), a copy of the foregoing was served via email to lead and backup counsel of record for Petitioner as follows:

Lead Counsel for Petitioner
Andrew M. Mason (Reg. No. 64,034) andrew.mason@klarquist.com
Back-Up Counsel for Petitioner
Cameron D. Clawson, Reg. No. 73,509 cameron.clawson@klarquist.com Sarah Jelsema, Reg. No. 70,804 sarah.jelsema@klarquist.com Frank Morton-Park, Reg. No. 80,750 frank.morton-park@klarquist.com
KLARQUIST SPARKMAN, LLP 121 SW Salmon Street, Suite 1600 Portland, Oregon, 97204 503-595-5300 (phone) 503-595-5301 (fax) Msft-Partec@klarquist.com

Dated: April 16, 2025

Respectfully submitted,

By: */ Michael F. Heim /*
Michael F. Heim (Reg. No. 32,702)
Attorney for Patent Owner
ParTec AG

CERTIFICATE OF COMPLIANCE

Pursuant to 37 C.F.R. § 42.24(d), the undersigned hereby certifies that this brief complies with the type-volume limitation of 37 C.F.R. § 42.24 because this brief contains 5,553 words.

Dated: April 16, 2025

Respectfully submitted,

By: */Michael F. Heim/*
Michael F. Heim (Reg. No. 32,702)
Attorney for Patent Owner
ParTec AG