

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

HULU, LLC,
Petitioner,

v.

SOUND VIEW INNOVATIONS, LLC,
Patent Owner.

Case IPR2018-00366
Patent 6,125,371

Before DEBRA K. STEPHENS, DANIEL J. GALLIGAN, and
JOHN A. HUDALLA, *Administrative Patent Judges*.

GALLIGAN, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
Inter Partes Review
35 U.S.C. § 318(a)

I. INTRODUCTION

In this *inter partes* review, Hulu, LLC (“Petitioner”) challenges the patentability of claims 1–3 and 8–10 of U.S. Patent No. 6,125,371 (Ex. 1201, “the ’371 patent”), which is assigned to Sound View Innovations, LLC (“Patent Owner”).

We have jurisdiction under 35 U.S.C. § 6. This Final Written Decision, issued pursuant to 35 U.S.C. § 318(a), addresses issues and arguments raised during the trial in this *inter partes* review. For the reasons discussed below, we determine that Petitioner has proven by a preponderance of the evidence that claims 1–3 and 8–10 of the ’371 patent are unpatentable. See 35 U.S.C. § 316(e) (“In an *inter partes* review instituted under this chapter, the petitioner shall have the burden of proving a proposition of unpatentability by a preponderance of the evidence.”).

A. Procedural History

On December 28, 2017, Petitioner filed a Petition (Paper 4 (“Pet.”)) requesting *inter partes* review of claims 1–3 and 8–10 of the ’371 patent, alleging obviousness under 35 U.S.C. 103(a)¹ over the combined teachings

¹ The Leahy-Smith America Invents Act, Pub. L. No. 112-29, 125 Stat. 284 (2011) (“AIA”), amended 35 U.S.C. §§ 102 and 103. Because the challenged claims of the ’371 patent have an effective filing date before the effective date of the applicable AIA amendments, we refer to the pre-AIA versions of 35 U.S.C. §§ 102 and 103.

of Bernstein² and Blake.³ Petitioner also filed a Motion for District Court-Type Claim Construction. Paper 3. Patent Owner filed a Preliminary Response. Paper 8 (“Prelim. Resp.”). On July 6, 2018, trial was instituted as to all challenged claims (claims 1–3 and 8–10) on the ground of unpatentability raised in the Petition—obviousness over the combination of Bernstein and Blake—and Petitioner’s Motion for District Court-Type Claim Construction was granted. Paper 11 (“Dec. on Inst.”), 37.

During the trial, Patent Owner filed a Response (Paper 21 (“PO Resp.”)), and Petitioner a Reply (Paper 25, “Pet. Reply”). With our authorization (Paper 26), Patent Owner filed a Sur-Reply (Paper 34, “PO Sur-Reply”).

An oral hearing was held on December 18, 2018, a transcript of which appears in the record. Paper 40 (“Tr.”).

B. Real Parties in Interest

Petitioner states: “Hulu, LLC, is the real party-in-interest. The following entities own ten percent or more of the stock of Petitioner, and may also be considered real parties-in-interest: The Walt Disney Company, 21st Century Fox, Comcast Corporation and Time Warner Inc.” Pet. 5. In Updated Mandatory Notices, Petitioner states the following: “Petitioner, Hulu, LLC, is and remains the real party-in-interest. The following entities

² PHILIP A. BERNSTEIN ET AL., CONCURRENCY CONTROL AND RECOVERY IN DATABASE SYSTEMS (1987). The full textbook (“the Bernstein textbook”) is located in Exhibit 1220. Exhibit 1203 (“Bernstein”) contains particular excerpts.

³ Russ Blake, *Optimizing Windows NT*, Microsoft Windows NT Resource Kit: For Windows NT Workstation and Windows NT Server Version 3.51 (Ex. 1219, “Blake”).

own ten percent or more of the stock of Hulu, LLC: AT&T, Inc., Comcast Corporation, Twenty-First Century Fox, Inc., and The Walt Disney Company. None of the foregoing parties listed as part-owners of Hulu, LLC controlled or funded this *inter partes* review proceeding (IPR), nor did they contribute to the preparation of this IPR in any way.” Paper 19, 1.

As real parties in interest, Patent Owner identifies itself and Sound View Innovation Holdings, LLC. Paper 10, 1.

C. Related Matters

As required by 37 C.F.R. § 42.8(b)(2), the parties identify various district court litigations involving the ’371 patent. Pet. 5–6; Paper 10, 1–2. The ’371 patent was at issue in IPR2017-00985, which we terminated at the request of the parties to that proceeding. The ’371 patent is also at issue in IPR2018-00017 and IPR2018-00096, in which we issue final written decisions concurrently with this Decision.

D. The ’371 Patent

The ’371 patent, titled “System and Method for Aging Versions of Data in a Main Memory Database,” generally describes systems and methods for managing versions of data records in a database to increase data capacity. Ex. 1201, [57], 2:55–62. The Background of the Invention Section of the ’371 patent explains that “[d]atabase systems typically include a database manager (‘DBM’) and a database (i.e., a data repository).” *Id.* at 1:13–15. “A DBM is a control application that supervises or manages interactions between application tasks and the database.” *Id.* at 1:15–17.

Figure 1 of the ’371 patent is reproduced below.

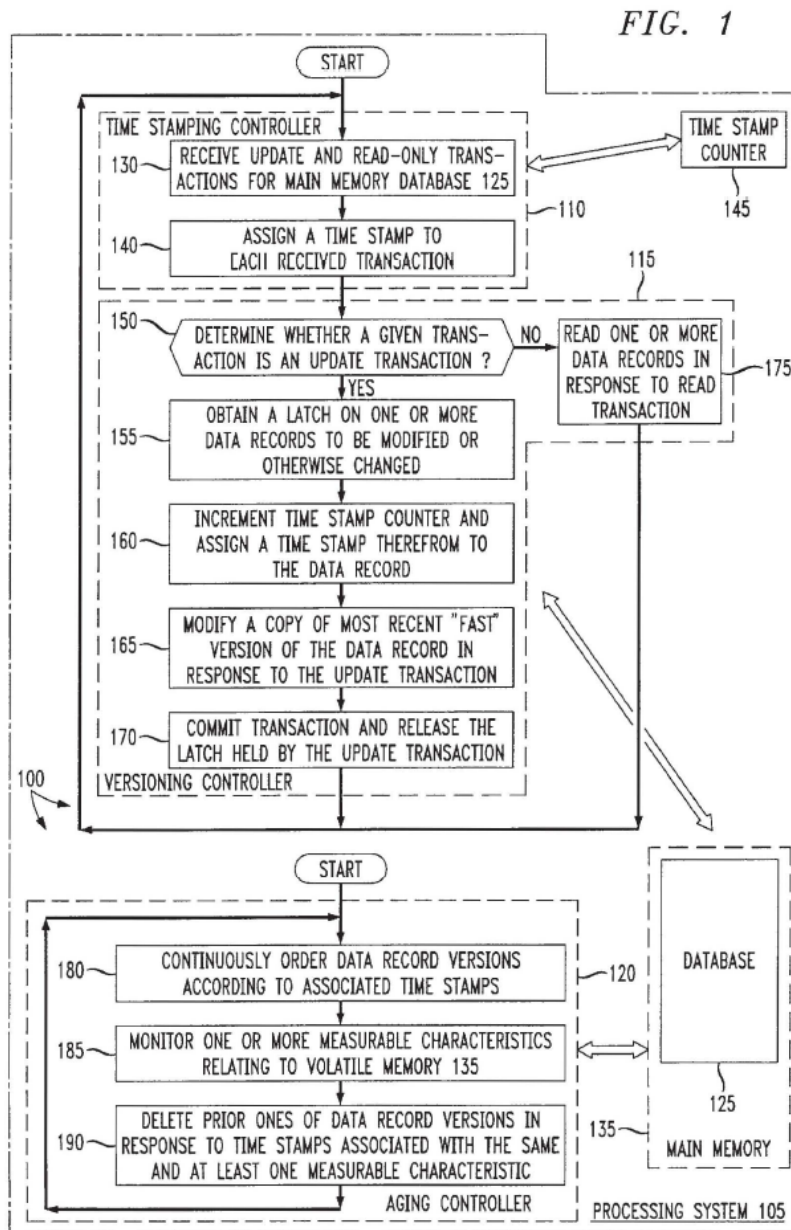


Figure 1 depicts a flow diagram of an exemplary method for controlling multi-versioned data records. Ex. 1201, 3:36–38. The '371 patent explains the following:

Version manager 100 may be software-based and executable by any suitably arranged processing system 105 (e.g., a computer, communications switch, etc.). Version manager 100 includes three controllers, namely, a time stamping controller 110, a versioning controller 115 and an aging controller 120.

Those skilled in the art should be familiar with the use of controllers in processing environments generally and, more specifically, with main memory databases. Controllers may be implemented in software, firmware, hardware, or some suitable combination of at least two of the three.

Id. at 4:47–57. Time stamping controller 110 receives update and read-only transactions for main memory database 125, step 130. *Id.* at 4:58–60. “In response, time stamping controller 110 assigns a time stamp to each received transaction, step 140.” *Id.* at 4:62–64. “[V]ersioning controller 115 determines whether a given transaction is an update transaction, decisional step 150.” *Id.* at 5:5–7.

If the transaction is an update transaction, . . . versioning controller 115 (1) obtains a “X” lock on one or more data records to be modified (or otherwise changed), step 155, (2) modifies a copy of the most recent “past” version of the data record in response to the update transaction, creating a new “current” or “successor” version, step 165 and (3) commits the transaction, at which time it increments time stamp counter 145, assigns a time stamp therefrom to the new “successor” versions of the updated data records and releases the “X” lock held by the update transaction, step 170.

Ex. 1201, 5:7–18. Aging controller 120 “monitors main memory database 125 to (1) continuously order (e.g., sort, arrange, etc.) multiple versions of ones of the data records according to their associated time stamps, step 180 and (2) monitor one or more measurable characteristics describing, relating to, or otherwise associated with a utilization or capacity of main memory 135, step 185.” *Id.* at 5:36–44. Aging controller 120 also deletes earlier versions of data records in response to the time stamp associated with those versions and at least one measurable main memory characteristic, step 190. *Id.* at 5:44–48.

E. Challenged Claims

Challenged claims 1–3 and 8–10 are reproduced below.

1. A processing system for use with a database of data records, said database stored in a memory, comprising:
 - a time stamping controller that assigns a time stamp to transactions to be performed on said database;
 - a versioning controller that creates multiple versions of ones of said data records affected by said transactions that are update transactions; and
 - an aging controller that monitors a measurable characteristic of said memory and deletes ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory.
2. The processing system as recited in claim 1 wherein said time stamp is generated as a function of a time stamp counter.
3. The processing system as recited in claim 2 wherein said system increments said time stamp counter.
8. A method of operating a processing system for use with a database of data records, said database stored in a memory, comprising the steps of:
 - assigning a time stamp to transactions to be performed on said database;
 - creating multiple versions of ones of said data records affected by said transactions that are update transactions;
 - monitoring a measurable characteristic of said memory; and
 - deleting ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory.
9. The method as recited in claim 8 wherein said time stamp is generated as a function of a time stamp counter.
10. The method as recited in claim 9 further comprising the step of incrementing said time stamp counter.

II. ANALYSIS

A. Level of Ordinary Skill in the Art

Citing the testimony of its declarant, Dr. Phillip Gibbons, Petitioner argues “[a] person of ordinary skill in the art would have had at least a bachelor’s degree in electrical engineering or computer science (or equivalent degree or experience) with at least two years of experience in the design and/or development of database and record management systems.” Pet. 8 (citing Ex. 1202 ¶¶ 10–12). Patent Owner does not provide a different assessment in the Patent Owner Response, but Patent Owner’s declarant, Dr. Mark Jones, provides the following testimony on this issue:

I generally agree with the level of ordinary skill described by Gibbons, except that at the time of the invention, a person of ordinary skill in the art would not have been able to devise algorithms or conduct original design or development of new database systems. Rather, a person of ordinary skill would have been able to implement database management software, such as multi-version concurrency control, based on algorithms and designs already provided to them. As will be discussed in more detail in connection with particular arguments, designing software such as multi-version concurrency control contains theoretical aspects (such as proof of correctness) and inventive aspects (such as practical garbage collection) that were beyond the abilities of the person of ordinary skill.

Ex. 2016 ¶ 21.

Thus, Dr. Jones does not disagree with Petitioner and Dr. Gibbons’s assessment but, rather, offers additional exceptions and qualifications, which we address in our analysis below. Because there is general agreement among the parties that Petitioner and Dr. Gibbons’s assessment of the level of ordinary skill is correct, we adopt Petitioner’s statement of the level of

ordinary skill in the art with the exception of the language “at least,” which introduces ambiguity. Thus, we determine that a person of ordinary skill in the art would have had a bachelor’s degree in electrical engineering or computer science (or equivalent degree or experience) with two years of experience in the design and/or development of database and record management systems.

B. Claim Interpretation

As noted above, Petitioner’s Motion for District Court-Type Claim Construction was granted because the ’371 patent expired on August 19, 2017. Dec. on Inst. 16, 37. In applying the district court-type construction standard, we are guided by the principle that the words of a claim “are generally given their ordinary and customary meaning,” as understood by a person of ordinary skill in the art in question at the time of the invention. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc) (citation omitted). “In determining the meaning of the disputed claim limitation, we look principally to the intrinsic evidence of record, examining the claim language itself, the written description, and the prosecution history, if in evidence.” *DePuy Spine, Inc. v. Medtronic Sofamor Danek, Inc.*, 469 F.3d 1005, 1014 (Fed. Cir. 2006) (citing *Phillips*, 415 F.3d at 1312–17). There is a “heavy presumption,” however, that a claim term carries its ordinary and customary meaning. *CCS Fitness, Inc. v. Brunswick Corp.*, 288 F.3d 1359, 1366 (Fed. Cir. 2002) (citation omitted).

Based on the record developed during trial, we address the interpretation of “measurable characteristic of said memory.”

1. “[M]easurable characteristic of said memory”

Claims 1 and 8 recite “monitor[ing] a measurable characteristic of said memory.” Patent Owner argues that “determining whether ‘memory is full’ does not necessarily entail monitoring a measurable characteristic” and that “‘memory is full’ is a logical determination, not a memory characteristic.” PO Resp. 56. According to Patent Owner, “[l]ogical events are **qualitative** assessments with a true/false outcome, such as whether a print job is completed, or whether there is disk error, or whether disk is full. In contrast, ‘measurable characteristics’ are **quantitative** assessments—hence, measurable.” PO Resp. 56 (citing Ex. 2016 ¶ 96).

In IPR2018-00017, which also involves the ’371 patent and the same Petitioner as this proceeding, Patent Owner argues that monitoring “logical events” does not teach monitoring a measurable characteristic of memory because “a measurable memory characteristic is a quantitative assessment, such as the percentage utilization of memory, e.g., 51%.” IPR2018-00017, Paper 21, 7. Patent Owner further argues that “‘disk full’ or ‘disk approaching threshold full’ logical events do not disclose the claims because they are logical events, which are immeasurable qualitative assessments.” IPR2018-00017, Paper 21, 52. Petitioner argues Patent Owner’s interpretation is unduly narrow. Pet. Reply 20–25. We agree with Petitioner.

Patent Owner’s interpretation of “measurable characteristic” is refuted by the specification of the ’371 patent, which states that

aging controller 120 determines a current value of L in response to one or more *measurable main memory characteristics*, step 200, *such as a current utilization or capacity of main memory*, a trend analysis of a utilization or capacity of main

memory over a time period (t), or any other applied mathematics- or statistics-based analysis, *including a comparison of any of the same with a threshold, ceiling/floor, limit, set point or the like.*

Ex. 1201, 6:46–53 (emphases added). According to the '371 patent, therefore, a comparison of a memory's current utilization or capacity to a threshold is within the scope of a "measurable main memory characteristic[]." Thus, Patent Owner's attempt to distinguish "logical events" and "qualitative assessments" from measurable characteristics of memory is contrary to the specification.

We determine that the phrase "measurable characteristic of said memory" does not require further construction.⁴

2. Remaining Terms

In the Decision on Institution, we adopted Petitioner's proposed constructions for the terms "update transactions" and "data record." Dec. on Inst. 17. The interpretation of these terms is not in dispute, and these constructions are not necessary to resolve the disputes at issue. We also determined that the limitations of claim 1 reciting "a time stamping controller," "a versioning controller," and "an aging controller" are not means-plus-function limitations. *Id.* at 18. The interpretation of these terms also is not in dispute. Based on the record before us, we determine that these terms and the remaining terms of the claims at issue do not require express constructions. *See, e.g., Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) ("[W]e need only construe terms 'that are in controversy, and only to the extent necessary to resolve the

⁴ In related litigation, the U.S. District Court for the Central District of California determined that this phrase did not require construction. IPR2018-00096, Ex. 1030, 13 (entered as Ex. 3002 in this proceeding).

controversy’” (quoting *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999))).

C. Principles of Law

A patent claim is unpatentable under 35 U.S.C. § 103(a) if the differences between the claimed subject matter and the prior art are such that the subject matter, as a whole, would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). The question of obviousness is resolved on the basis of underlying factual determinations including (1) the scope and content of the prior art; (2) any differences between the claimed subject matter and the prior art; (3) the level of ordinary skill in the art; and (4) any secondary considerations, if in evidence.⁵ *Graham v. John Deere Co.*, 383 U.S. 1, 17–18 (1966).

D. Obviousness over Bernstein and Blake (Claims 1–3 and 8–10)

Petitioner contends claims 1–3 and 8–10 of the ’371 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over the combination of Bernstein and Blake. Pet. 25–50.

1. Overview of Bernstein (Ex. 1203)

Exhibit 1203 contains excerpts from a textbook on database systems, titled *Concurrency Control and Recovery in Database Systems*. Petitioner asserts that “Bernstein qualifies as prior art to the claims of the ’371 patent under 35 U.S.C. § 102(b) (pre-AIA).” Pet. 15. Petitioner introduces

⁵ Patent Owner has not presented evidence of secondary considerations as to any of the challenged claims.

additional evidence that the Bernstein textbook was published more than one year before the August 19, 1997, filing date of the '371 patent. *See* Exs. 1206 (Bernstein textbook excerpts, including title page and date stamp (April 27, 1987) from Library of Congress), 1207 (Bernstein textbook excerpts, including stamp from the University of Michigan Libraries and date annotation (1987)). During the trial, Patent Owner did not contest the status of Bernstein as a prior art printed publication under 35 U.S.C. § 102(b). Based on the evidence of record, we determine Bernstein is a prior art printed publication under 35 U.S.C. § 102(b).

Bernstein explains that “[a] *database* consists of a set of named *data items*.” Ex. 1203, 2.⁶ “Each data item has a *value*.” *Id.* “A *database system (DBS)* is a collection of hardware and software modules that support commands to access the database, called *database operations* (or simply *operations*).” *Id.* (footnote omitted). For example, a “Read(*x*)” operation “returns the value stored in data item *x*,” and a “Write(*x*, *val*)” operation “changes the value of *x* to *val*.” *Id.*

Bernstein teaches a number of techniques for addressing concurrent access problems. *Id.* at 1. In particular, Bernstein explains that “[w]hen two or more transactions execute concurrently, their database operations execute in an *interleaved* fashion. That is, operations from one program may execute in between two operations from another program. This interleaving can cause programs to behave incorrectly, or *interfere*, thereby leading to an inconsistent database.” *Id.* at 11. One of the techniques described in Bernstein to provide concurrency control is referred to as “multiversion

⁶ Our citations are to the page numbers of the Bernstein reference itself, rather than to the page numbers of Exhibit 1203.

concurrency control.” *Id.* at 143. “In a multiversion concurrency control algorithm, each Write on a data item x produces a new copy (or *version*) of x .” *Id.*

“The benefit of multiple versions for concurrency control is to help the scheduler avoid rejecting operations that arrive too late.” Ex. 1203, 143. Bernstein explains that, with multiversion concurrency control, “each transaction has a unique timestamp” and that “[e]ach operation carries the timestamp of its corresponding transaction.” *Id.* at 153; *see also id.* at 5 (“transactions that write into the database (called *update transactions* or *updaters*)”). For example, as noted above, each Write operation produces a new copy or version of x (*id.* at 143), and the new version is “labeled by the timestamp of the transaction that wrote it.” *Id.* at 153.

Bernstein acknowledges that “[a]n obvious cost of maintaining multiple versions is storage space. To control this storage requirement, versions must periodically be purged or archived.” *Id.* at 143–44. Bernstein explains that versions may be purged or archived when the system has run out of storage space. In particular, Bernstein discloses the following:

Eventually, the scheduler will run out of space for storing intervals, or the [data manager] will run out of space for storing versions. At this point, old versions and their corresponding intervals must be deleted. To avoid incorrect behavior, it is essential that versions be deleted from oldest to newest.

Id. at 154.

2. Overview of *Blake* (Ex. 1219)

Petitioner asserts that “*Blake* qualifies as prior art under 35 U.S.C. § 102(b) (pre-AIA) because it was published on March 1, 1995, more than one year before the earliest patent application filing date on the face of the ’371 patent.” Pet. 18. In its Preliminary Response, Patent Owner argued

Petitioner failed to show Blake was publicly accessible before the August 19, 1997, filing date of the '371 patent. Prelim. Resp. 57–59. With its Reply, Petitioner submits the Declaration of James L. Mullins, Ph.D. Ex. 1227. Dr. Mullins testifies that Blake “was publicly available and accessible no later than the end of the November 1996.” Ex. 1227 ¶ 36. During the trial, Patent Owner did not contest the status of Blake as a prior art printed publication under 35 U.S.C. § 102. November 1996 is not more than one year prior to the August 19, 1997, filing date of the '371 patent and, thus, would not qualify Blake as a prior art printed publication under 35 U.S.C. § 102(b), as asserted by Petitioner in the Petition. Nevertheless, based on Dr. Mullins’s testimony in rebuttal to Patent Owner’s argument, we determine Blake is prior art printed publication under 35 U.S.C. § 102(a).

Blake describes a “Performance Monitor” tool that allows monitoring various “objects” in a computer system, including physical disks and memory. Ex. 1219, 21–22. Blake explains: “Each object has a set of *counters* defined for it. An object’s counters record the activity level of the object.” Ex. 1219, 21. One such “counter” is “% Free Space,” which “is the ratio of the free space available on the logical disk unit to the total usable space provided by the selected logical disk drive.” Ex. 1219, 22, 428.

3. Independent Claims 1 and 8

Claim 1 is directed to “[a] processing system for use with a database of data records, said database stored in a memory,” and claim 8 is directed to “[a] method of operating a processing system for use with a database of data records, said database stored in a memory.” Petitioner argues Bernstein teaches the preambles of claims 1 and 8. Pet. 25–27, 48–49 (citing Ex. 1203, 2, 17; Ex. 1202 ¶¶ 48–51, 70–71). Bernstein explains that “[a]

database consists of a set of named *data items*.” Ex. 1203, 2. Bernstein describes a database system (DBS) as “a collection of hardware and software modules that support commands to access the database” and further discloses that “the DBS executes on a *centralized* computer system.” Ex. 1203, 2, 17. We find that Bernstein teaches the preambles of claims 1 and 8.

a. Time stamping controller / assigning a time stamp

Claim 1 recites “a time stamping controller that assigns a time stamp to transactions to be performed on said database,” and claim 8 recites the step of “assigning a time stamp to transactions to be performed on said database.” Bernstein discloses that, “[a]s for all [timestamp ordering (“TO”)] schedulers, each transaction has a unique timestamp, denoted $ts(T_i)$. Each operation carries the timestamp of its corresponding transaction. Each version is labeled by the timestamp of the transaction that wrote it.” Ex. 1203, 153, *quoted in* Pet. 28; *see* Ex. 1202 ¶ 52. Petitioner contends Bernstein discloses that “the ‘transaction man[a]ger’ (TM) performs the actual timestamp assignment.” Pet. 28–30 (citing Ex. 1203, 17, 85). We are persuaded because Bernstein discloses the following: “Usually, TMs assign timestamps to transactions. If there is only one TM in the entire system, then it can easily generate timestamps by maintaining a counter. To generate a new timestamp, it simply increments the counter and uses the resulting value.” Ex. 1203, 85. As Petitioner notes (Pet. 26), Bernstein describes that a database system (DBS) is “a collection of hardware and software modules that support commands to access the database” (Ex. 1203, 2). Bernstein describes the transaction manager as one of four modules in the DBS, along with a scheduler, a recovery manager, and a cache manager.

Ex. 1203, 17. We find that Bernstein teaches “a time stamping controller” because Bernstein discloses its DBS as a collection of hardware and software modules and also discloses a module that assigns time stamps, and the ’371 patent states that “[c]ontrollers may be implemented in software, firmware, hardware, or some suitable combination of at least two of the three.” Ex. 1201, 4:55–57.

Based on the evidence before us, we find that Bernstein teaches “a time stamping controller,” as recited in claim 1, and the step of “assigning a time stamp,” as recited in claim 8.

b. Versioning controller / creating multiple versions of data records

Claim 1 further recites “a versioning controller that creates multiple versions of ones of said data records affected by said transactions that are update transactions,” and claim 8 further recites the step of “creating multiple versions of ones of said data records affected by said transactions that are update transactions.” According to Petitioner, Bernstein teaches that “when a ‘**Write**’ operation is to be performed on an item in the database, the software generates a new version of the item.” Pet. 32 (citing Ex. 1203, 143). In particular, Bernstein discloses that, “[i]n a multiversion concurrency control algorithm, each Write on a data item *x* produces a new copy (or *version*) of *x*. The [data manager (“DM”)] that manages *x* therefore keeps a list of versions of *x*, which is the history of values that the DM has assigned to *x*.” Ex. 1203, 143. As discussed above, Bernstein discloses that the operation “Write(*x*, *val*) changes the value of *x* to *val*.” Ex. 1203, 2, *quoted in* Pet. 33; *see* Ex. 1202 ¶ 57. Although these passages of Bernstein refer to an “operation,” rather than a “transaction,” we credit the testimony of Petitioner’s declarant that this difference is immaterial. Ex. 1202 ¶ 58;

see Ex. 1203, 2 (referring to “*database operations* (or simply *operations*)”), 5 (referring to “transactions that write into the database” as “*update transactions* or *updaters*”).

Based on the evidence before us, we find that Bernstein teaches “a versioning controller,” as recited in claim 1, and the step of “creating multiple versions,” as recited in claim 8.

c. Aging controller that monitors a measurable characteristic of said memory / monitoring a measurable characteristic

Claim 1 further recites “an aging controller that monitors a measurable characteristic of said memory,” and claim 8 further recites the step of “monitoring a measurable characteristic of said memory.” Petitioner argues that Bernstein alone renders obvious this subject matter and also that Blake, in combination with Bernstein, teaches this limitation. Pet. 35–43. We address each of these contentions in turn.

i. Bernstein alone

In its contentions as to this limitation, Petitioner relies on the following passage of Bernstein:

Eventually, the scheduler will run out of space for storing intervals, or the DM will run out of space for storing versions. At this point, old versions and their corresponding intervals must be deleted. To avoid incorrect behavior, it is essential that versions be deleted from oldest to newest.

Ex. 1203, 154, *quoted in* Pet. 35. Petitioner argues:

Although the passage from Bernstein above does not expressly state that the system “**monitor[s] a measurable characteristic of said memory**,” it would have been obvious to a person of ordinary skill in the art that the disclosures in Bernstein disclose this step. ([Ex. 1202] ¶ 53.) Bernstein explains that “[a]n obvious cost of maintaining multiple versions is storage space. To control this storage requirement, versions

must be periodically purged or archived.” ([Ex. 1203,⁷] 143–44.) To this end, the passage quoted above indicates that Bernstein can detect when it has “**run out of space**” for storing intervals or data items, and can respond to that condition by deleting older versions. (*Id.*, p.154.) Therefore, it would have been obvious that this functionality discloses the ability to ascertain the amount of available space for storing items, *i.e.*, monitoring a measurable characteristic of the system’s memory. ([Ex. 1202] ¶ 53.) Otherwise, the system could not determine when it has run out of space. (*Id.*) Moreover, determining the amount of free space or memory was a standard and well-known feature. (*Id.*)

Pet. 36; see Ex. 1202 ¶ 61.

Patent Owner argues Bernstein does not teach monitoring a measurable characteristic of memory because, “[f]irst, determining whether ‘memory is full’ does not necessarily entail monitoring a measurable characteristic” and “[s]econd, whether or not ‘memory is full’ is a logical determination, not a memory characteristic.” PO Resp. 56. As discussed above in the section addressing claim interpretation, the ’371 patent provides examples of “measurable main memory characteristics,” including

a current utilization or capacity of main memory, a trend analysis of a utilization or capacity of main memory over a time period (t), or any other applied mathematics- or statistics-based analysis, including a comparison of any of the same with a threshold, ceiling/floor, limit, set point or the like.

Ex. 1201, 6:47–53 (emphasis added). A determination that memory is full is a determination as to the current utilization of the memory. Thus, whether or not it is a “logical determination,” as Patent Owner argues in its second point, is irrelevant because such a determination is still within the scope of a

⁷ Petitioner cites “*Id.*, pp.143-44.” Pet. 36. Although the previous citation is to the Gibbons declaration, based on the context of the statement and the citation, we understand Petitioner’s citation to refer to Bernstein.

measurable characteristic according to the specification of the '371 patent.

In support of its first reason, Patent Owner argues “[t]here are numerous ways to determine when the system has ‘run out of memory’ that have nothing to do with monitoring a measurable characteristic,” including that “the system could respond to error codes when writes are repeatedly failing, or when a request to the operating system to allocate memory for additional copies or intervals is rejected.” PO Resp. 56 (citing Ex. 2016 ¶ 95). Patent Owner further argues that “[a] logical event such as ‘disk full’ may be monitored and discovered through numerous methods as discussed above, such as failure to allocate memory, without requiring any ‘monitoring of a measurable memory characteristic.’” PO Resp. 57–58 (citing Ex. 2016 ¶ 98).

Patent Owner’s arguments about responding to error codes and a failure in allocating memory ignore Bernstein’s express disclosure. Bernstein does not state that, eventually, there will be errors in allocating memory, which could result from less space being available than requested. Rather, Bernstein states that, “[e]ventually, the scheduler will *run out of space* for storing intervals, or the DM will *run out of space* for storing versions. At this point, old versions and their corresponding intervals must be deleted.” Ex. 1203, 154 (emphases added). Bernstein, therefore, discloses “run[ning] out of space” (Ex. 1203, 154), which teaches a measure of the “current utilization” of memory—i.e., fully utilized. See Ex. 1201, 6:48. And Bernstein’s disclosure that versions “must be deleted” upon “run[ning] out of space” teaches *monitoring* a measurable characteristic of the memory by determining that space in memory has “run out” such that a particular action (deleting versions) is taken in response.

Based on the evidence before us, we find Bernstein teaches “an aging controller that monitors a measurable characteristic of said memory,” as recited in claim 1, and the step of “monitoring a measurable characteristic of said memory,” as recited in claim 8, according to the specification of the ’371 patent, which identifies “current utilization” of memory as a “measurable main memory characteristic[.]” Ex. 1201, 6:47–53.

ii. Bernstein and Blake

In addition, Petitioner relies on Blake in combination with Bernstein for the “monitor[ing] a measurable characteristic” limitations of claims 1 and 8, arguing the following:

It is possible that the patent owner might take a narrow position on the “monitoring” limitation in order to assert that Bernstein alone does not sufficiently disclose the claimed “monitoring” feature, or could argue for a different interpretation of the teachings of Bernstein. For this reason, this Petition also cites to the **Blake** reference. Blake clearly and explicitly discloses the claimed monitoring feature and, as explained above, is readily combinable with the database system in Bernstein.

Pet. 37.

Petitioner relies on Blake’s “Performance Monitor” for an additional teaching of “monitoring a measurable characteristic of said memory.”

Pet. 37–38. Figure 2.12 of Blake is reproduced below.

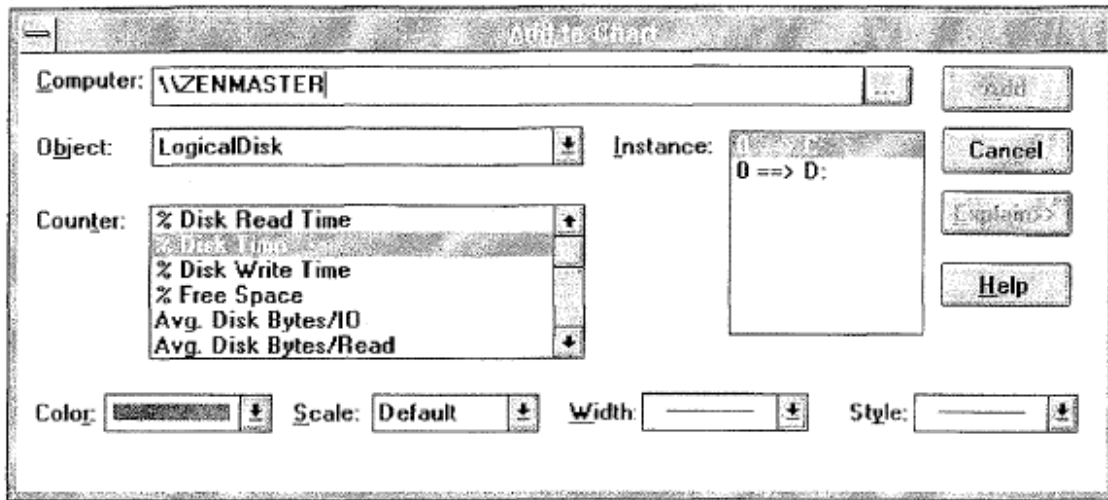


Figure 2.12 shows an “Add to Chart dialog box” for the Performance Monitor. Ex. 1219, 22. As explained above, Blake’s “Performance Monitor” tool monitors various “objects” in a computer system, including physical disks and memory, where “[e]ach object has a set of *counters*,” which “record the activity level of the object.” Ex. 1219, 21–22. One such “counter” depicted in Figure 2.12 is “% Free Space,” which “is the ratio of the free space available on the logical disk unit to the total usable space provided by the selected logical disk drive.” Ex. 1219, 22, 428.

Patent Owner argues that Blake’s Performance Monitor teachings, in combination with Bernstein, do not teach monitoring a measurable characteristic of memory, as recited in claims 1 and 8. PO Resp. 49–52. In particular, Patent Owner argues that, if a person of ordinary skill in the art were implementing a database in Windows NT, which is the operating system Blake discusses, that person “would have either bypassed the Windows file system, or would have defined the database as a large file.” PO Resp. 49 (citing Ex. 2017, 45:5–12). According to Patent Owner, if the Windows file system were bypassed, the Performance Monitor would see

the entire space allocated to the database as used space and, therefore, would not see what part of the database itself is actually filled with database records. PO Resp. 50–51. Patent Owner argues that allocating the database as a large file with a fixed file size would have the same result because, from the Performance Monitor’s perspective, the space allocated for the file would be considered used space regardless of the extent to which space within the file is consumed with database records. PO Resp. 51.

As an initial matter, we find Blake teaches monitoring a measurable characteristic of memory because it discloses measuring the remaining amount of free memory with the “% Free Space” counter, which “is the ratio of the free space available on the logical disk unit to the total usable space provided by the selected logical disk drive.” Ex. 1219, 21–22, 428. This is consistent with the examples provided by the ’371 patent of “measurable main memory characteristics,” including

a current utilization or capacity of main memory, a trend analysis of a utilization or capacity of main memory over a time period (t), or any other applied mathematics- or statistics-based analysis, including a comparison of any of the same with a threshold, ceiling/floor, limit, set point or the like.

Ex. 1201, 6:47–53 (emphases added). We do not agree with Patent Owner that combining Blake with Bernstein undermines Blake’s express teachings of monitoring a measurable characteristic of memory. Patent Owner attempts to defeat the teachings of Blake through bodily incorporation of those teachings into Bernstein; however, the Court of Appeals for the Federal Circuit has

consistently held . . . that “[t]he test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference; nor is it that the claimed invention must be expressly suggested in any

one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art.”

MCM Portfolio LLC v. Hewlett-Packard Co., 812 F.3d 1284, 1294 (Fed. Cir. 2015) (quoting *In re Keller*, 642 F.2d 413, 425 (CCPA 1981)). Here, Blake expressly teaches monitoring a measurable characteristic of memory, such as through its “% Free Space” counter. Ex. 1219, 21–22, 428.

Based on the evidence before us, we find Blake alone, or taken in combination with the disclosures of Bernstein discussed above in our analysis of this limitation, teaches “an aging controller that monitors a measurable characteristic of said memory,” as recited in claim 1, and the step of “monitoring a measurable characteristic of said memory,” as recited in claim 8.

d. Aging controller that deletes versions/deleting versions

Claim 1 recites that the aging controller “deletes ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory,” and claim 8 recites the step of “deleting ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory.” Petitioner argues that Bernstein teaches deleting old versions of data items and their corresponding intervals to recover storage space. Pet. 43–44 (citing Ex. 1203, 143–144, 154). Further, because the deletions are based on the age of the version, as well as the storage space available, Petitioner argues Bernstein alone and the combination of Bernstein and Blake teach that such deletions are “in response to said time stamp and said measurable characteristic.” Pet. 43–45 (citing Ex. 1203, 85–86, 154, 161;

Ex. 1219, 45–46); *see* Ex. 1202 ¶¶ 63–65; *see also* Ex. 1203, 153 (“largest,” i.e., latest, time stamp), 161 (“smallest,” i.e., earliest time stamp). As noted by Petitioner, Bernstein discloses that “[t]o avoid incorrect behavior, it is essential that versions be deleted from oldest to newest.” Ex. 1203, 154, *quoted in* Pet. 44. As further noted by Petitioner, Bernstein discloses that, “[s]ince timestamps increase monotonically with time and are unique, if a transaction lives long enough it will eventually have the smallest timestamp (i.e., will be the oldest) in the system.” Ex. 1203, 85–86, *quoted in* Pet. 44. Moreover, Petitioner argues that these deletions “increase a capacity of said memory.” Pet. 45 (citing Ex. 1203, 154, 161); *see* Ex. 1202 ¶ 66. In addition to its arguments relying on Bernstein alone, Petitioner also argues these limitations of claims 1 and 8 would have been “obvious in view of Blake because . . . the alert function in Blake’s Performance Monitor may be used in conjunction with Bernstein to delete old data when the amount of free space on a logical disk, or the amount of available bytes in memory, fall below a user-defined threshold.” Pet. 43–44 (citing Ex. 1219, 45–46; Ex. 1202 ¶ 63).

According to Patent Owner, “[i]dentifying which versions can be deleted and recovering the space associated with them” is “a concept known as ‘garbage collection.’” PO Resp. 20 (citing Ex. 2013, 50:14–25). The “delet[ing]” limitations of claims 1 and 8 are allegedly directed to such “garbage collection.” *See* PO Resp. 58. Patent Owner argues

Bernstein does not disclose any garbage collection algorithm for its [multiversion timestamp ordering (MVTO)] method, and the [person of ordinary skill in the art] would not have found it obvious or reasonably feasible to combine Bernstein’s MVTO method and Blake to design a practical garbage collection algorithm practicing the Claims.

PO Resp. 18–19 (citing Ex. 2016 ¶¶ 39–40). Patent Owner further asserts the following:

Petitioners’ conclusory assertion that the “in response to” limitation of the Claim is disclosed by Bernstein’s statement that “old versions” must be deleted because memory “will eventually run out of space,” Pet. at 43, is nonsensical. It is based on a superficial hindsight reading of Bernstein’s statement, divorced from realities that the [person of ordinary skill in the art] would have understood at the time. Bernstein is not stating that old versions must be deleted *in response to* memory running out of space; rather, Bernstein is merely stating that old versions should be deleted *because* memory will “eventually” run out. Bernstein is stating a need for garbage collection but, for MVTO, is not providing a **solution** for that need. Without a practical means for garbage collection, a multi-versioning method for database systems is not useful or practical.

PO Resp. 24–25 (citing Ex. 2016 ¶¶ 48–49).

Patent Owner’s argument ignores the express disclosure of Bernstein. Bernstein does not state, and Petitioner does not allege that Bernstein states, “that ‘old versions’ must be deleted because memory ‘will eventually run out of space.’” See PO Resp. 24. Bernstein discloses the following:

Eventually, the scheduler will run out of space for storing intervals, or the DM will run out of space for storing versions. *At this point*, old versions and their corresponding intervals must be deleted. To avoid incorrect behavior, it is essential that versions be deleted from oldest to newest.

Ex. 1203, 154 (emphasis added). Petitioner asserts “Bernstein discloses that the deletion of items occurs when Bernstein has ‘run out of space.’” Pet. 43 (citing Ex. 1203, 154). We agree with Petitioner because Bernstein says “[a]t this point,” i.e., at the point that the scheduler has “run out of space for storing intervals” or the data manager has “run out of space for storing versions,” “old versions and their corresponding intervals must be deleted.”

Ex. 1203, 154. We, therefore, disagree with Patent Owner's statement that "Bernstein is not stating that old versions must be deleted *in response to* memory running out of space" (PO Resp. 24) because it is contrary to the express disclosure of Bernstein cited by Petitioner.

Patent Owner also argues Bernstein does not teach deleting "in response to said time stamp." PO Resp. 58–60. In particular, Patent Owner argues Bernstein "does not even suggest that in MVTO, the method would determine the oldest version to delete by performing any operation on, or relating to, the time stamp." PO Resp. 60.

We disagree. In describing MVTO, Bernstein discloses that, "[a]s for all TO schedulers, each transaction has a unique timestamp, denoted $ts(T_i)$. Each operation carries the timestamp of its corresponding transaction. Each version is labeled by the timestamp of the transaction that wrote it."

Ex. 1203, 153. Bernstein further describes timestamps as follows: "Since timestamps increase monotonically with time and are unique, if a transaction lives long enough it will eventually have the smallest timestamp (i.e., will be the oldest) in the system." Ex. 1203, 85–86. Thus, when Bernstein states that, "[t]o avoid incorrect behavior, it is essential that versions be deleted from oldest to newest" (Ex. 1203, 154), it is clear that the oldest version is determined based on the timestamp, i.e., the oldest version is the version with the smallest timestamp.

Patent Owner also argues that,

even if the combined teachings of Bernstein and Blake disclosed deleting ones of said multiple versions of said ones of said data records in response to said time stamp, or performing such deleting in response to said measurable characteristic of memory, they do not teach performing such deleting in response to **both** of these things, as required by the claims.

PO Resp. 60.

We disagree. Bernstein discloses that “the scheduler will run out of space for storing intervals, or the DM will run out of space for storing versions” and that “[a]t this point, old versions and their corresponding intervals must be deleted. To avoid incorrect behavior, it is essential that versions be deleted from oldest to newest.” Ex. 1203, 154. Thus, Bernstein teaches deleting versions in response to space running out and also in response to the time stamp, which determines which version is the oldest. *See* Ex. 1203, 85–86 (disclosing that, “if a transaction lives long enough it will eventually have the smallest timestamp (i.e., will be the oldest) in the system”).

Based on the evidence before us, we find Bernstein teaches “an aging controller that . . . deletes ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory,” as recited in claim 1, and the step of “deleting ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory,” as recited in claim 8. Furthermore, we find that the deletions in Bernstein “increase a capacity of said memory.” Ex. 1202 ¶ 66.

We also are persuaded by Petitioner’s contention that the combination of Bernstein and Blake teaches “deleting ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory.” *See* Pet. 43–44. In particular, Petitioner relies on Blake’s disclosures of providing alerts and taking action in response to such alerts. Pet. 40, 43–44

(citing Ex. 1219, 45–46; Ex. 1202 ¶ 63). For example, Blake discloses providing an alert when the value of the “% Free Space” counter falls below 10 percent. Ex. 1219, 45–46. Blake states that “[y]ou can designate a program to be run either the first time or every time each different alert is triggered.” Ex. 1219, 46. Thus, we find Blake teaches taking some action in response to a measurable characteristic of memory, and we find that this teaching in combination with Bernstein’s disclosures regarding deleting old versions teaches “delet[ing] ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory.”

e. Rationale to combine

Petitioner argues that a person of ordinary skill in the art would have had reason to combine the teachings of Bernstein and Blake. Pet. 38–42; *see* Ex. 1202 ¶¶ 78–85. In particular, as noted above, Bernstein teaches deleting older versions only *after* the scheduler or the data manager has run out of storage space. Ex. 1203, 154; *see also* Pet. 40–41; *see* Ex. 1202 ¶¶ 82–84. Petitioner notes that this “may require that the entire database system be suspended until the deletion process frees up the needed space.” Pet. 41. Petitioner argues that Blake’s teachings of providing various alerts when certain conditions occur, such as when the amount of free space falls below a particular threshold (Ex. 1219, 45–46), “would have improved the database system of Bernstein by allowing it to proactively delete old versions before the database runs out of storage space—thereby allowing the database to consistently maintain a minimal level of space (as defined by the user-provided threshold).” Pet. 41–42 (citing Ex. 1219, 45–46; Ex. 1202 ¶ 84). Petitioner asserts that the combination of Bernstein and Blake

would have predictably resulted in the database system of Bernstein with the ability to detect when the amount of “Free Space” in a “Logical Disk,” or the amount of “Available Bytes” in “Memory,” have fallen below a user-specified threshold, to which the database system could respond by automatically deleting old versions.

Pet. 38–39 (citing Ex. 1202 ¶ 78). Petitioner, therefore, relies on Blake’s monitoring of disk storage space and argues that combining this functionality with Bernstein would have been advantageous to proactively alert Bernstein’s data manager of impending disk space shortages rather than waiting until the data manager “run[s] out of space for storing versions” (Ex. 1203, 154) to free up disk space. Pet. 38–42.

Patent Owner argues that “[a]ll of the purported motivations proffered by Petitioner presume that the [person of ordinary skill in the art] would attempt to devise a garbage collection algorithm for MVTO,” which, according to Patent Owner, Bernstein does not disclose. PO Resp. 35. Patent Owner argues the following:

Petitioner does not . . . address the underlying question: Why would the [person of ordinary skill in the art] attempt to use Bernstein’s MVTO, which would have required the [person of ordinary skill in the art] to devise a new garbage collection algorithm with no expectation of success, where Bernstein does not disclose such an algorithm and where one was not commonly known in the art, while disregarding other single- and multi-version concurrency control methods with known, practical algorithms?

PO Resp. 35.

We disagree with Patent Owner’s premise that Bernstein does not disclose a “garbage collection” scheme for MVTO. As we discuss above, according to Patent Owner, “[i]dentifying which versions can be deleted and recovering the space associated with them” is “a concept known as ‘garbage

collection.” PO Resp. 20 (citing Ex. 2013, 50:14–25). Bernstein discloses deleting the oldest versions when space has run out, i.e., “garbage collection” commensurate with the scope of claims 1 and 8. Ex. 1203, 154.

Patent Owner argues that a person of ordinary skill in the art would not have been motivated to combine the teachings of Bernstein and Blake without having access to the Performance Monitor software itself, rather than just the Blake publication, and Patent Owner also notes that such software could not be the basis of an *inter partes* review as it would not be “prior art consisting of patents or printed publications.” PO Resp. 43–45 (quoting 35 U.S.C. § 311(b)). Patent Owner further argues that Blake pertains to the Windows NT operating system but that the person of ordinary skill in the art would have chosen to implement the system in an operating system such as UNIX instead. PO Resp. 46–49. According to Patent Owner, “Windows NT was a relatively new operating system” in 1997, and the person of ordinary skill in the art would have known “there would be a substantial learning curve for developing reliable, efficient, high-performance applications on Windows NT” and, therefore, would not have chosen Windows NT as the operating system in which to implement a “complex database management software that utilized multi-versioning.” PO Resp. 47–48.

We disagree with these arguments because, as discussed above, bodily incorporation of teachings from one reference into another reference is not the test for obviousness. *See MCM Portfolio*, 812 F.3d at 1294. That Blake describes functionality and features of a particular operating system (Windows NT) does not foreclose considering Blake’s teachings in combination with Bernstein. “Rather, the test is what the combined

teachings of the references would have suggested to those of ordinary skill in the art.” *Id.* (internal quotation and citation omitted). Blake “is prior art for all that it teaches,” *Beckman Instruments, Inc. v. LKB Produkter AB*, 892 F.2d 1547, 1551 (Fed. Cir. 1989), and it teaches monitoring a measurable characteristic of memory, as explained above. Ex. 1219, 45–46.

We are persuaded Petitioner’s articulated reasons to combine the teachings of Bernstein and Blake have a rational underpinning, and specifically, we find a person of ordinary skill in the art would have been motivated to combine Blake’s teachings of monitoring disk storage space with Bernstein to proactively alert Bernstein’s data manager of impending disk space shortages rather than waiting until the data manager “run[s] out of space for storing versions.” See Ex. 1202 ¶¶ 82–84. We agree with Dr. Gibbons that “[m]aintaining [a] minimum level of space, in turn, would have avoided the need to suspend the database to delete old versions.” Ex. 1202 ¶ 84.

f. Reasonable expectation of success

Patent Owner argues a person of ordinary skill in the art would not have had a reasonable expectation of success in “combin[ing] Bernstein’s MVTO method and Blake to design a *practical* garbage collection algorithm practicing the Claims.” PO Resp. 18–19 (emphasis added); see generally PO Resp. 19–34. Throughout its discussion of this issue, Patent Owner repeatedly argues or implies that reasonable expectation of success requires a “practical” solution. See, e.g., PO Resp. 20 (“A correct and practical garbage collection scheme is not merely the arbitrary deletion of old versions.”), 20 (“Absent some inventive, intelligent methodology, a brute-force approach to searching through the database to find multiple versions of

the data items in order to identify and delete unnecessary versions was not (and is not) a practical approach.”), 23 (“[A] practical multi-versioning method in a database system requires an effective solution other than ‘brute force’ to the problem of garbage collection.”), 23 (“A practical garbage collection algorithm for MVTO (which Bernstein describes as its first method, and on which Petitioners have relied for the disclosure of the ‘in response to’ limitation) was not ordinary knowledge in the art as of 1997.”), 25 (“Without a practical means for garbage collection, a multi-versioning method for database systems is not useful or practical.”), 26 (alleging a “lack of disclosure in Bernstein of a practical garbage collection algorithm for its MVTO method”), 30 (alleging that “a practical method of garbage collection for MVTO was not within the common knowledge or ability of the” person of ordinary skill in the art).

We discern no legal requirement for a particular combination to yield a “practical” solution. The Court of Appeals for the Federal Circuit has stated that “[t]he reasonable expectation of success requirement refers to the likelihood of success in combining references *to meet the limitations of the claimed invention.*” *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1367 (Fed. Cir. 2016) (emphasis added); *see also Allergan, Inc. v. Sandoz Inc.*, 726 F.3d 1286, 1292 (Fed. Cir. 2013) (“[T]he person of ordinary skill need only have a reasonable expectation of success of developing the claimed invention.”). Independent claims 1 and 8 do not recite any requirement for a “practical” garbage collection process. Rather, claims 1 and 8 recite “delet[ing] ones of said multiple versions of said ones of said data records in response to said time stamp and said measurable characteristic thereby to increase a capacity of said memory.” As explained

above, we find claims 1 and 8 encompass the deletion of versions as taught in Bernstein.

In its Response, Patent Owner attempts to distinguish the subject matter of the challenged claims from the teachings of Bernstein and Blake based on features from the specification of the '371 patent, such as the use of a version list entry and a time stamp parameter "L." See PO Resp. 37–43. These features, however, are not recited in the challenged claims but, rather, are recited in dependent claims 5–7 and 12–14 of the '371 patent, which are not challenged in the Petition. In its Sur-Reply, Patent Owner argues that it is not attempting to import limitations from the specification into the claims but, instead, "is simply pointing out that the [person of ordinary skill in the art] would have needed **some** practical garbage collection algorithm in order to have a reasonable expectation of success in implementing Bernstein's MVTO." PO Sur-Reply 9. As the Supreme Court stated in *KSR*, "[w]hat matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103." *KSR*, 550 U.S. at 419. Claims 1 and 8 do not require a "practical" garbage collection algorithm.

Patent Owner's arguments and Dr. Jones's testimony also touch on the level of ordinary skill in the art. According to Patent Owner, "[b]ecause of the difficulties and lack of solutions described above, the [person of ordinary skill in the art] would have no reasonable expectation of success in designing a practical garbage collection algorithm for MVTO as of 1997." PO Resp. 27. Patent Owner further argues that, "if a person in 1997 had invented a practical garbage collection algorithm for MVTO concurrency control, that person would **not** have been considered as having merely **ordinary** skill. He would have been someone of extraordinary skill—or an

inventor.” PO Resp. 27–28. Furthermore, as noted above in section II.A, Dr. Jones testifies that

a person of ordinary skill in the art would not have been able to devise algorithms or conduct original design or development of new database systems. Rather, a person of ordinary skill would have been able to implement database management software, such as multi-version concurrency control, based on algorithms and designs already provided to them. As will be discussed in more detail in connection with particular arguments, designing software such as multi-version concurrency control contains theoretical aspects (such as proof of correctness) and inventive aspects (such as practical garbage collection) that were beyond the abilities of the person of ordinary skill.

Ex. 2016 ¶ 21. It may be true that it would take a person of extraordinary skill in the art to devise a *practical* garbage collection algorithm, but that question is not before us. The claims simply do not recite such a requirement. “Rather, the person of ordinary skill need only have a reasonable expectation of success of *developing the claimed invention*.” *Allergan*, 726 F.3d at 1292 (emphasis added).

We are persuaded that a person of ordinary skill in the art would have had a reasonable expectation of success in arriving at the invention recited in claims 1 and 8 based on Bernstein alone and based on the combined teachings of Bernstein and Blake. As to Bernstein alone, as we discuss above, we find Petitioner has shown that Bernstein teaches all of the limitations of claims 1 and 8, including the deleting limitation. Bernstein discloses deleting versions from oldest to newest when space has run out for storing intervals or versions. Ex. 1203, 154. As discussed above, Patent Owner’s arguments that reasonable expectation of success requires a “practical” solution are not commensurate with the scope of claims 1 and 8. Dr. Gibbons testifies that “a person of ordinary skill in the art would not

have required disclosures any more detailed than the disclosures in Bernstein to apply the prior art teachings in the manner described in this Declaration.” Ex. 1202 ¶ 88. This testimony, which we credit, supports a finding that a person of ordinary skill in the art would have had a reasonable expectation of success of developing the *claimed invention* based on the teachings of Bernstein. Thus, based on Petitioner’s persuasive showing, we find a person of ordinary skill in the art would have had a reasonable expectation of success in arriving at the subject matter recited in claims 1 and 8 based on Bernstein alone.

We also find that a person of ordinary skill in the art would have had a reasonable expectation of success combining the teachings of Bernstein and Blake to arrive at the subject matter recited in claims 1 and 8. *See* Pet. 38–42; Ex. 1202 ¶¶ 78–84. In particular, we credit Dr. Gibbons’s testimony that the combination

would have predictably resulted in the database system of Bernstein with the ability to detect when the amount of “Free Space” in a “Logical Disk,” or the amount of “Available Bytes” in “Memory,” have fallen below a user-specified threshold, to which the database system could respond by automatically deleting old versions.

Ex. 1202 ¶ 78.

g. Conclusion as to claims 1 and 8

In summary, we find Bernstein teaches all of the limitations of claims 1 and 8, and we also find the combination of Bernstein and Blake teaches all of the limitations of claims 1 and 8. In addition, we find that a person of ordinary skill in the art would have had a reasonable expectation of success in arriving at the invention recited in claims 1 and 8 based on Bernstein alone and based on the combined teachings of Bernstein and

Blake. We also find that a person of ordinary skill in the art would have been motivated to combine the teachings of Bernstein and Blake. Having considered the full record developed during trial, we conclude Petitioner has shown, by a preponderance of the evidence, that the subject matter of claims 1 and 8 would have been obvious under 35 U.S.C. § 103(a) based on Bernstein alone and based on the combined teachings of Bernstein and Blake.

4. Claims 2, 3, 9, and 10

Claims 2 and 9 depend, respectively, from claims 1 and 8 and recite that “said time stamp is generated as a function of a time stamp counter.” Claim 3 depends from claim 2 and recites that “said system increments said time stamp counter.” Claim 10, which depends from claim 9, similarly recites “incrementing said time stamp counter.” Although Patent Owner does not provide specific arguments regarding claims 2, 3, 9, and 10 beyond its arguments for independent claims 1 and 8, the burden remains on Petitioner to demonstrate unpatentability. *See Dynamic Drinkware LLC, v. Nat’l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015). For the reasons explained below, we determine Petitioner has met its burden.

Petitioner argues Bernstein teaches the limitations of claims 2, 3, 9, and 10. Pet. 47–48, 50 (citing Ex. 1203, 85). In particular, Bernstein discloses: “Usually, TMs [(transaction managers)] assign timestamps to transactions. If there is only one TM in the entire system, then it can easily generate timestamps by maintaining a counter. To generate a new timestamp, it simply increments the counter and uses the resulting value.” Ex. 1203, 85. According to Petitioner, this passage of Bernstein “satisfies the limitations of claims 2 and 3 because it discloses that the system

described in Bernstein maintains ‘**a counter**,’ and assigns new timestamps by incrementing the timestamp counter and using the just-incremented counter value.” Pet. 48. Noting the similarity of claims 2 and 9 and that of claims 3 and 10, Petitioner argues claims 9 and 10 would have been obvious for the same reasons as claims 2 and 3. Pet. 50. We are persuaded by Petitioner’s argument, and we find Bernstein teaches the additional limitations recited in claims 2, 3, 9, and 10.

Having considered the full record developed during trial, we conclude Petitioner has shown, by a preponderance of the evidence, that the subject matter of claims 2, 3, 9, and 10 would have been obvious under 35 U.S.C. § 103(a) based on Bernstein alone and based on the combined teachings of Bernstein and Blake.

III. WEIGHT OF TESTIMONY

Patent Owner argues “the Board should not give weight to the testimony of Petitioner’s purported expert Dr. Gibbons” for various reasons, including his alleged “insubstantial analysis and understanding of the claimed invention or even the references, as well as his rubber-stamping of a mostly *verbatim* copy of another declarant’s earlier-filed declaration as his own.” PO Resp. 8–12. Patent Owner was afforded the opportunity to, and did, cross-examine Dr. Gibbons regarding the testimony he provided in his declaration in this proceeding and in IPR2018-00017. *See* Ex. 2017 (transcript of Gibbons deposition in this proceeding); Ex. 2013 (transcript of Gibbons deposition in IPR2018-00017). Patent Owner highlights some of that testimony as allegedly undermining Dr. Gibbons’s credibility. For example, Patent Owner quotes the following question and answer:

[Q] So is it fair to say that sitting here now you don't know what the inventors of the '371 patent meant by referring to logical aging?

[A] Correct.

Ex. 2013, 101:7–11 (objection omitted), *quoted in* PO Resp. 9. The phrase “logical aging,” however, is not recited in the challenged claims. Although Dr. Gibbons's inability to explain “logical aging” may indicate his unfamiliarity at the time of his deposition with certain aspects of the '371 patent specification, we do not see this as a reason to completely ignore his testimony regarding the subject matter recited in the challenged claims.

Patent Owner also highlights passages from the following exchange in Dr. Gibbons's deposition:

[Q] I asked you a few moments ago if you fully understand Bernstein's third mixed method of concurrency control. Do you remember that?

A Yes.

Q And you testified that you haven't reviewed Bernstein so you can't answer my question basically, correct?

[A] I testified that the – as I sit here now, I would need to review the multiversion mixed method section of Bernstein before I could describe how it works. It is true that at the time I submitted my declaration I had similarly not described -- determined in my head exactly how it worked.

Q Now, at the time you submitted your declaration, had you analyzed why a designer in 1997 would choose Bernstein's mixed method as opposed to its multiversion timestamp ordered or vice-versa? And if you had not analyzed it, you can just state so. That's okay.

A At the time of my declaration, I was looking for aspects of Bernstein that taught what was in the claims, and so I latched on to the multiversion timestamp ordering as an example of a concurrency control algorithm that teaches the claims. That did

not rule out in my head that there might be other multiversion concurrency control algorithms in the paper that also teach the claims. So there was no need to further investigate which of these multiversion methods was -- had advantages or disadvantages over any of the other ones.

Q I don't think that answered my question, so I'll repeat my question. At the time you submitted your declaration, had you analyzed whether a designer or a POSITA in 1997 would choose Bernstein's third mixed method of multiversion concurrency control as opposed to the multiversion timestamp ordered or vice-versa?

[A] At the time of my declaration, I had not considered which between those two would have -- would have been considered -- what the pros and cons between the two would have been for a designer back in 1997.

Ex. 2017, 26:21–29:2 (objections omitted); *see* PO Resp. 11 (citing Ex. 2017, 27:9–14, 28:13–29:2). Patent Owner argues that this testimony shows that Dr. Gibbons did not “have even a basic understanding of Bernstein's disclosures relating to multi-versioning concurrency control.” PO Resp. 11.

We disagree that this testimony undermines Dr. Gibbons's opinions because Petitioner and Dr. Gibbons rely on Bernstein's MVTO disclosures, *not* Bernstein's other multiversion techniques, to show obviousness of the claims, a point made by Patent Owner in its Response. PO Resp. 4 (“That a practical garbage collection algorithm for Bernstein's MVTO was not commonly known as of the Patent's invention date is also highlighted by the fact that, in contrast to Bernstein's lack of disclosure with respect to its discussion of MVTO, Bernstein does indeed disclose practical garbage collection algorithms for both of the other multi-version concurrency control methods that it discusses—however, the Petition does not suggest that either

of these other methods renders the claimed inventions obvious.”). Furthermore, the examining attorney’s question about “whether a designer or a POSITA in 1997 would choose Bernstein’s third mixed method of multiversion concurrency control as opposed to the multiversion timestamp ordered or vice-versa” (Ex. 2017, 28:12–18) is irrelevant to the issues in this case. Dr. Gibbons testified that he “latched on to the multiversion timestamp ordering as an example of a concurrency control algorithm that teaches the claims.” Ex. 2017, 28:2–5; *see KSR*, 550 U.S. at 419 (“What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under § 103.”). That Bernstein discloses other techniques in addition to MVTO does not diminish the teachings of MVTO relied upon by Petitioner and Dr. Gibbons.

Therefore, we do not agree with Patent Owner that Dr. Gibbons’s testimony should be given no weight. Rather, we have weighed the evidence before us in arriving at our conclusions, as laid out in our analysis above.

IV. CONCLUSION

For the reasons discussed above, we determine Petitioner has proven, by a preponderance of the evidence, that claims 1–3 and 8–10 of the ’371 patent are unpatentable under 35 U.S.C. § 103(a) over the teachings of Bernstein alone and also over the combined teachings of Bernstein and Blake.

V. ORDER

Accordingly, it is:

ORDERED that claims 1–3 and 8–10 of U.S. Patent No. 6,125,371 have been shown to be unpatentable; and

FURTHERED ORDERED that, because this is a Final Written Decision, parties to the proceeding seeking judicial review of the Decision must comply with the notice and service requirements of 37 C.F.R. § 90.2.

IPR2018-00366
Patent 6,125,371

PETITIONER:

David Cavanaugh
Scott Bertulli
Arthur Shum
WILMERHALE
david.cavanaugh@wilmerhale.com
scott.bertulli@wilmerhale.com
Arthur.shum@wilmerhale.com

PATENT OWNER:

Kenneth Weatherwax
Parham Hendifar
Nathan Lowenstein
LOWENSTEIN & WEATHERWAX LLP
weatherwax@lowensteinweatherwax.com
hendifar@lowensteinweatherwax.com
lowenstein@lowensteinweatherwax.com