

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

DELPHIX CORP.,
Petitioner,

v.

ACTIFIO, INC.,
Patent Owner.

Case IPR2015-01678
Patent 8,299,944 B2

Before THOMAS L. GIANNETTI, DAVID C. McKONE, and
ROBERT L. KINDER, *Administrative Patent Judges*.

KINDER, *Administrative Patent Judge*.

DECISION
Denying Institution of *Inter Partes* Review
37 C.F.R. § 42.108

Delphix Corp. (“Petitioner”) filed a Petition pursuant to 35 U.S.C. §§ 311–19 to institute an *inter partes* review of claims 1–4 (all claims) of U.S. Patent No 8,299,944 B2, issued on October 30, 2012 (Ex. 1001, “the ’944 patent”). Paper 1 (“Pet.”). Actifio, Inc. (“Patent Owner”) filed a Preliminary Response. Paper 6 (“Prelim. Resp.”). We have jurisdiction under 35 U.S.C. § 314 and 37 C.F.R. § 42.4(a).

To institute an *inter partes* review, we must determine that the information presented in the Petition shows “a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition.” 35 U.S.C. § 314(a). Having considered both the Petition and the Preliminary Response, we are not persuaded that Petitioner has demonstrated a reasonable likelihood that it would prevail in showing the unpatentability of any of claims 1–4 of the ’944 patent on any alleged ground. Accordingly, we do not institute an *inter partes* review of any claim on the record before us.

I. BACKGROUND

A. The ’944 patent (Ex. 1001)

The ’944 patent is titled “System and Method for Creating Deduplicated Copies of Data Storing Non-Lossy Encodings of Data Directly in a Content Addressable Store.” The “invention relates generally to data management, data protection, disaster recovery and business continuity.” Ex. 1001, 1:44–48. The ’944 patent discloses improvements for storing data in a redundant fashion. *Id.* at 2:4–31.

The ’944 patent describes a unified Data Management Virtualization System (“DMVS”) for data management activities. *See id.* at 3:20–31. The

claims of the '944 patent relate to certain aspects of this system, such as “storing deduplicated images in which a portion of the image is stored in encoded form directly in a hash table” and storing hashes pointing to the stored data. *Id.* at Abstract, 1:44–48, 34:5–46. The disclosed DMVS abstracts physical storage resources into a series of virtualized data protection storage pools.

The capacity optimized storage pool, also called the content addressable store (“CAS”), “provides a way of storing common subsets of different objects only once, where those common subsets may be of varying sizes.” *Id.* at 25:61–65. The CAS stores data as content addressable objects. *Id.* at 27:56–59, 26:19–22. The system generates a data structure called a “handle” and associates the handle with each content addressable object (data object) to be stored. *Id.* at 27:56–59. The “handle” includes information about the data object, including the size of the data object, and a “hash” of the data object. *See id.* at 27:59–63. The handle relates to the claimed “hash structure” as further explained below. *See* Pet. 7–8 (“A ‘hash structure’ in the claims refers to what the specification describes as a ‘handle’ . . .”).

Figure 12, depicted below, “shows how the handle for a content addressed object is generated.” Ex. 1001, 27:56–57.

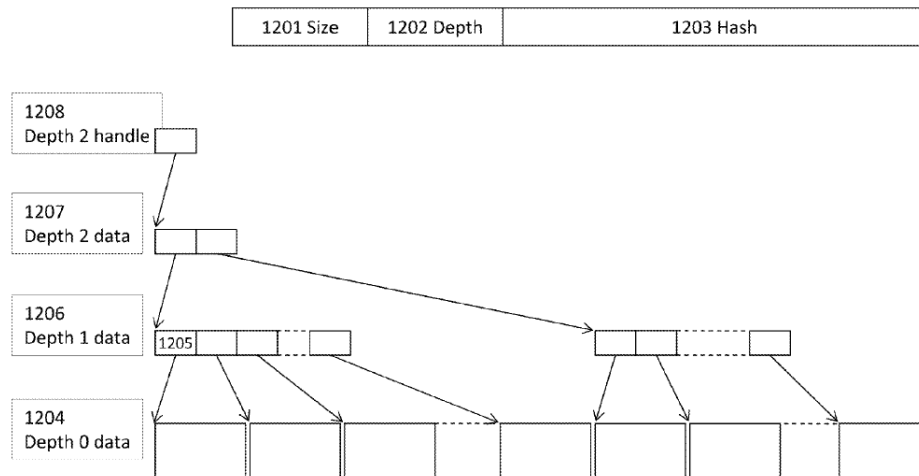


FIG. 12

Referring to Figure 12, the data object manager references all content addressable objects with a content addressable handle. Each handle “is made up of three parts,” including a hash of the data object (1203), so that it can be readily identified. *Id.* at 27:59–63.

The hash in the handle can be either: (i) the fixed length hash key of an object the handle points to generated via a hashing algorithm such as SHA-1 that “generates a secure cryptographic hash with a uniform distribution and a probability of hash collision near enough zero,” or (ii) a non-lossy encoding of the data segment itself. *Id.* at 27:2–12, claim 1. The non-lossy encoding of the data segment is stored as a hash value when the segment fits within the field for containing a hash. *Id.* at 27:64–28:4, claim 1.

The hash (1203) of the data object “optionally includes a tag indicating that the hash field is a non-lossy encoding of the underlying data.” *Id.* at 27:62–64. The Specification further discloses “[t]he tag indicates the

encoding scheme used, such as a form of run-length encoding (RLE) of data used as an algorithmic encoding if the data chunk can be fully represented as a short enough RLE.” *Id.* at 27:64–28:1.

B. Illustrative Claim

Claim 1, reproduced below, is the only independent claim and is illustrative of the claims at issue:

1. A method of storing deduplicated images in which a portion of the image is stored in encoded form directly in a hash table, the method comprising:

[1A] organizing unique content of each data object as a plurality of content segments and storing the content segments in a data store;

[1B] for each data object, creating an organized arrangement of hash structures, wherein each structure, for a subset of the hash structures, includes a field to contain a hash signature for a corresponding content segment and is associated with a reference to the corresponding content segment,

[1C] wherein the logical organization of the arrangement represents the logical organization of the content segments as they are represented within the data object;

[1D] receiving content to be included in the deduplicated image of the data object;

[1E] generating a new hash structure for the received content;

[1F] determining if the received content may be encoded using a predefined non-lossy encoding technique and in which the encoded value would fit within the field for containing a hash signature;

[1G] if so, placing the encoding in the field of the new hash structure instead of placing a hash signature for the received content in the field, and marking the new hash structure to indicate that the field contains encoded content for the

deduplicated image, so that the received content is not stored since the received content can be derived from the encoding;

[1H] if not, generating a hash signature for the received content and placing the hash signature in the field of the new hash structure and placing the received content in a corresponding content segment in said data store if it is unique.

Ex. 1001, 34:6–39 (bracketed numbering and paragraphing added).

C. Related Proceedings

Petitioner and Patent Owner identify a related litigation in the District of Massachusetts involving the '944 patent: *Actifio, Inc. v. Delphix Corp.*, Case No. 1:14-cv-13247-DJC. Pet. 1; Paper 5.

D. Real Party-in-Interest

Petitioner certifies that the real party-in-interest for this Petition is Delphix Corporation. Pet. 1.

E. References

Petitioner relies on the following references:

Reference		Date	Exhibit
Patterson	US 7,143,251 B1	Nov. 28, 2006	Ex. 1002
Zhu	US 6,928,526 B1	Aug. 9, 2005	Ex. 1003
Terao	US 2004/0199570 A1	Oct. 7, 2004	Ex. 1004
Li	US 7,689,633 B1	Sept. 15, 2004 (filed) Mar. 30, 2010 (issued)	Ex. 1005

F. Grounds Asserted

Petitioner challenges the patentability of claims 1–4 of the '944 patent on the following two grounds (Pet. 3):

Reference(s)	Basis	Claims Challenged
Patterson (incorporating Zhu), and Terao	§ 103(a)	1, 3, and 4
Patterson, (incorporating Zhu), Terao, and Li	§ 103(a)	2

II. ANALYSIS

A. Claim Construction

In an *inter partes* review, claim terms in an unexpired patent are interpreted according to their broadest reasonable construction in light of the specification of the patent in which they appear. *See* 37 C.F.R. § 42.100(b); *In re Cuozzo Speed Techs., LLC*, 793 F.3d 1268, 1275–79 (Fed. Cir. 2015), *cert. granted sub nom., Cuozzo Speed Techs., LLC v. Lee*, No. 15–446, 2016 WL 205946 (U.S. Jan. 15, 2016).

Petitioner proposes the construction of three terms (Pet. 7–8), and Patent Owner does not challenge these proposed claim constructions (Prelim. Resp. 29). Only the term “hash structure” is relevant for our analysis, and we construe it as follows.

The term “hash structure” appears in claim 1. The Specification describes the claimed “hash structure” as a handle. Pet. 7–8, Prelim. Resp. 6. *See also* Ex. 1001, 27:55–59 (“The data object manager references all content addressable objects with a content addressable handle.”). Petitioner contends that the term “hash structure” should be construed as “*a structure for holding a value generated via a hash function.*” Pet. 8 (emphasis added). Petitioner relies on the Specification and its explanation that a hash is generated via a hashing algorithm such as SHA-1. *Id.* (citing Ex. 1001,

27:1–12). Petitioner also quotes two dictionaries providing definitions for the term “hash.” *Id.*

On this record, we adopt Petitioner’s proposed construction as the broadest reasonable interpretation. Thus, “hash structure” is construed to mean “*a structure for holding a value generated via a hash function.*”

B. Obviousness of Claim 1, 3, and 4 Over Patterson, Zhu, and Terao

Petitioner contends that claims 1, 3, and 4 would have been obvious over Patterson, Zhu,¹ and Terao. Pet. 3. Having considered the arguments and evidence before us, for the reasons set forth below, Petitioner does not make a sufficient showing that the subject matter of any of claims 1, 3, and 4 would have been obvious over Patterson, Zhu, and Terao.

1. Patterson

Patterson relates to “a data storage system and method that efficiently eliminate[s] redundancy.” Ex. 1002, 1:5–8. As depicted in Figure 3 below, Patterson receives a segmented data stream (300) and uses a hash table in determining whether each segment in the stream has been stored previously (306). Ex. 1002, Fig. 3, 4:35–38, 5:62–6:22.

¹ Petitioner argues that Zhu is incorporated by references into Patterson or, alternatively, that certain aspects of Zhu would have been obvious to incorporate into embodiments disclosed in Patterson. Pet. 14.

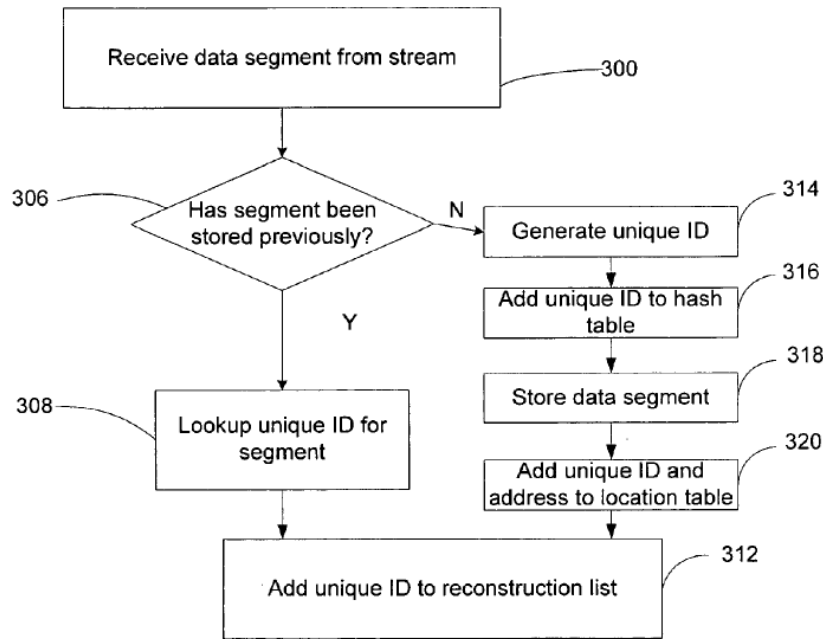


Figure 3 of Patterson is a flowchart illustrating the write process used for storing a segmented data stream. Ex. 1002, 1:60–61.

If the system determines that an identical data segment has not been stored previously, the data segment is added to data segment storage and associated with a unique identifier that is added to a hash table (steps 306, 314, 316, 318). *Id.* at Fig. 3, 4:33–58, 6:3–9, 6:19–22. If there is an identical stored data segment (306), the incoming segment is not added to data segment storage, but instead the unique identifier for the data segment is looked up (308) and added to a reconstruction list (312). *Id.* at 4:46–48, 6:14–19. The reconstruction list is a sequence of unique identifiers stored in the same order in which the data segments were received in the data stream and later used to reconstruct the original data stream with the contents of stored data segments. *Id.* at 3:14–16, 4:4–22, 4:46–57, 6:3–9, 6:14–19.

Patterson describes various types of unique identifiers that may be used. *Id.* at 4:23–24. For example, the unique identifiers stored in the

reconstruction lists can be derived from “content derived summaries” or fingerprints such as hashes. *Id.* at 2:37–54; 5:31–47. Patterson also discloses that the unique identifiers can be the data segments themselves. *Id.* at 4:31–32. A data segment’s unique identifier is associated with the memory location of the data segment that it represents. *Id.* at 2:45–54, 3:60–4:3, 5:13–47, Figs. 2B and 5A. For example, referring to Figure 5A, a unique identifier in the reconstruction list is an index into a table that includes a corresponding fingerprint of the data segment and a memory location where the data segment is stored. *See id.* at 5:12–19. Patterson explains that the system retrieves the data segments stored at the memory addresses associated with each unique identifier found in the reconstruction list and assembles the data segments in that order. *Id.* at 4:58–5:2, 7:34–59, Fig. 7.

Patterson discloses how data can be looked up using unique identifiers. Ex. 1002, 3:30–5:2; 5:13–59. In one embodiment, a hash table illustrated in Figure 2A is used to determine if a data segment is already stored by comparing the data segment’s hash value to the entries in the table. Ex. 1002, 3:30–59; 4:33–57. The second table illustrated in Figure 2B is used to locate a data segment during a read operation. *Id.* at 3:60–4:3. In a second embodiment, referring to Figures 5A, 5B, 6 and 7, Patterson discloses using a single table instead of two tables for writing and looking up data segment locations. *See id.* at 5:13–7:59, 5:3–12 (“in some embodiments, a combined table is used for both the reading and the writing of data”).

2. Zhu

Petitioner contends that Zhu is incorporated by reference into Patterson and thus, Patterson “includes Zhu’s description as if it was itself recited in Patterson.” Pet. 9. Patent Owner does not challenge this contention.

Zhu is entitled “Efficient Data Storage System.” Zhu’s “system determines whether a data segment has been stored previously” and if so, “an identifier for the previously stored data segment is returned.” Ex. 1003, Abstract. Zhu receives a data stream from a backup server or other data source and divides the stream into segments. *Id.* at 2:65–3:2. The system passes these segments to an ID generator that generates a segment ID to be associated with each of the data segments from the input data stream. *Id.* at 2:65–3:2, 3:15–19. Zhu discloses that these segment IDs can be hash signatures generated by a cryptographic hash function such as the MD5 algorithm, a fingerprint, or “the ID is the segment data itself.” *Id.* at 3:24–33.

Once a data segment’s segment ID is generated, the system sends the segment ID and other metadata to a segment redundancy check engine that determines whether the segment has been previously stored in a content addressable segment database. *Id.* at 3:34–4:27, 5:12–30, 7:36–8:47. If the engine determines that an identical data segment has not previously been stored, the data segment is added to a segment database. *Id.* at 8:1–4, 8:27–40.

Zhu discloses that data segments also may be stored in compressed form. For example, Zhu describes that the data segments may be encoded using a lossless compression algorithm such as Lempel-Ziv. *Id.* at 5:1–5.

Referring to Figure 3 of Zhu, the segment data is compressed and stored in its associated content segment (i.e., data store (304)). *See id.* at Fig. 3, 4:63–5:11.

3. Terao

Terao relates to an information processing system for holding and providing information as processing objects. Ex. 1004 ¶ 4. Terao discloses that a server computer may divide a piece of incoming data into a sequence of data fragments and calculate a characteristic value for each data fragment. *Id.* at Abstract, ¶¶ 8, 9, 12, 31. Terao describes that a characteristic value may be a hash value. *Id.* ¶ 29. Terao discloses comparing the size of each data fragment with the size of that fragment’s hash value. *Id.* ¶¶ 12, 38. If the data fragment is larger than its characteristic value, the server stores the data fragment. *Id.* ¶¶ 12, 39. If the characteristic value is larger than the data fragment, Terao describes that the server may simply use the data fragment as its own “characteristic value.” *Id.* ¶¶ 12, 37, 38. In this disclosed embodiment, the “data fragment per se”—i.e., the raw data fragment—is used as the characteristic value. *Id.* ¶ 12. Thus, Terao’s data fragment also is stored as raw data, regardless of whether it is larger or smaller than the characteristic value. *Id.* ¶¶ 12, 37–39.

4. Claim 1

Petitioner contends that the combination of Patterson, Zhu, and Terao teaches or suggests each limitation of claim 1. Pet. 18–32. Petitioner supports its contentions with citations to the references and with the testimony of Prashant Shenoy, Ph.D. Ex. 1009. Petitioner alleges that both Patterson and Zhu disclose hash-based deduplication of stored data. Pet. 9. Petitioner identifies Patterson’s “unique identifier” as the claimed “hash

signature” and Patterson’s “reconstruction list” as the “hash structure” of the claims. *Id.* at 23–24. According to Petitioner, “Patterson further discloses that each unique identifier in the reconstruction list is associated with a reference to the corresponding data segment, specifically a memory location where the data segment is stored.” *Id.* at 23 (“a unique identifier in the reconstruction list is an index into a table that includes a corresponding fingerprint of the data segment and a memory location where the data segment is stored”). Petitioner relies on Zhu for its disclosure that a segment ID can be the segment data itself or a digital signature (cryptographic hash or a fingerprint) generated from a hash function computed using the segment data. *Id.* at 10, 24. Petitioner also relies on Zhu for its disclosure that the data segments can be stored in compressed form using a lossless compression encoding technique. *Id.* at 19. Petitioner contends it would have been obvious to store such a compressed encoding in the reconstruction list (hash structure) of Patterson in further view of Terao. According to Petitioner, Terao discloses use of a data segment as an identifier in a hash structure if it is smaller than the output of a given hash function used to generate an identifier of a data segment. *Id.* at 28. Thus, Petitioner relies on Terao as teaching the claimed step of determining if the encoded value would fit within the field for containing a hash signature. *See* step 1F below.

Patent Owner challenges whether the combination of Patterson, Zhu, and Terao teach or suggest limitations found in steps 1B, 1F, and 1G of claim 1. Prelim. Resp. 38–46. We focus our analysis on limitations found in steps 1F and 1G for the reasons set forth below.

i. Steps 1F and 1G of Claim 1

Step 1F requires, “determining if the received content may be encoded using a predefined non-lossy encoding technique and in which the encoded value would fit within the field for containing a hash signature.” Ex. 1001, 34:24–27. If the conditions of step 1F are met, step 1G further requires “placing the encoding in the field of the new hash structure . . . and marking the new hash structure to indicate that the field contains encoded content for the deduplicated image so that the received content is not stored since the received content can be derived from the encoding.” *Id.* at 34:28–34.

ii. Petitioner’s Contentions for Steps 1F and 1G

Petitioner relies on Zhu’s compression technique integrated into Patterson’s embodiments, as well as portions of Terao, as meeting the limitations of steps 1F and 1G. Pet. 27–31. Petitioner alleges “[i]t would be obvious to one skilled in the art that the data segments in the Patterson embodiments could also be compressed with a non-lossy encoding technique such as Lempel-Ziv.” Pet. 28. Petitioner then concludes “it would be obvious to one skilled in the art that storing data segments in Patterson’s embodiments in the compressed form as described in Zhu would render the deduplicated storage system even more efficient.” Pet. 28.

Petitioner further relies on Terao as teaching a comparison of an encoded data segment to a unique identifier to determine if the encoded data segment would fit into a field in the reconstruction list. *Id.* at 29. Using Terao’s comparison technique, Petitioner contends

that a losslessly encoded data segment itself, as described in Zhu incorporated in Patterson, can be stored in Patterson’s reconstruction list rather than the unique identifier of that data segment if the encoded data segment is smaller than the unique

identifier calculated from a hash of the segment, and thus would fit into a field in the reconstruction list that is designed to store larger unique identifiers, because both Patterson and Zhu describe that an identifier can be the data segment itself and Patterson explicitly explains the advantage of having “shorter” identifiers.

Pet. 29. Based on this proposed combination, Petitioner argues

there must be a way for the system to distinguish such entries [(i.e., the stored encoded data segment)] from those that include a hash based identifier of the data segment so that the system can determine whether the data segment needs to be retrieved from storage using the segment identifier or can be obtained directly from the list of segment identifiers.

Id. at 30 (citing the expert declaration of Prashant Shenoy, Ph.D (“Shenoy Decl.”) ¶ 81).

Petitioner contends that the marking requirement of limitation 1G is met by “an indicator as to whether the unique identifier is the encoded data segment itself or a value derived from a hash function of the data segment” *Id.* (citing Shenoy Decl. ¶ 81).

iii. Patent Owner’s Contentions for Steps 1F and 1G

Patent Owner contends that the references do not teach or suggest the limitations of steps 1F and 1G, either alone or in combination. Prelim. Resp. 1, 12, 14, 15, 20, 21, 33, 38–46. Patent Owner argues that “in the only disclosed embodiment of Zhu that actually uses compression, the compressed data is stored in fields and structures separate from any hash structures.” *Id.* Likewise, Patent Owner argues that “nothing in Zhu discloses storing the encoding *in the hash structure* and there are no disclosures in Patterson or Zhu that would teach a [person of ordinary skill in the art] to do this.” *Id.* at 39. According to Patent Owner, “[n]either

Petitioner nor its expert explain why a [person of ordinary skill in the art] would try to encode data and then store it specifically in a hash structure.” *Id.* Patent Owner further argues Terao teaches away from encoding data because it only discloses storing raw, non-encoded data and any storage gained from compression would be padded away. *Id.* at 39, 25 (citing Ex. 1004, ¶ 33 (“the tail is preferably padded with a predetermined value”)).

Patent Owner further contends that the combination of prior art fails to teach or suggest marking the new hash structure to indicate that the field contains encoded content for the deduplicated image as required by step 1G of claim 1. *Id.* at 42–45. According to Patent Owner, because none of the references discloses placing encoded data in a hash structure, the references also do not and cannot disclose marking the new hash structure to indicate that the field contains encoded content as claimed. *Id.* at 43. Patent Owner reasons that “it would not have been obvious to include a marking in the hash structure to indicate the presence of such an encoding, because there was no need in those references to differentiate a hash structure that stores a hash signature from a hash structure that stores an encoding.” *Id.* at 44. Patent Owner also argues Terao cannot support marking as claimed because Terao teaches that the received content is always stored in raw form. *Id.* Thus, a person of ordinary skill in the art would have had no reason to modify Patterson and Zhu in light of Terao to mark the hash structure to indicate that it has encoded content. *Id.*

Finally, Patent Owner argues that the Petition fails to identify the differences between the prior art references and the claims of the ’944 patent (*id.* at 29–32), lacks an adequate reason or motivation to combine or modify

the references as proposed (*id.* at 23–28), and uses impermissible hindsight to make modifications not contemplated by the references (*id.* at 26, 28).

iv. Analysis of Steps 1F and 1G

Petitioner has not persuasively established that the proposed combination of prior art teaches or suggests the requirements of steps 1F and 1G. We agree with Patent Owner that neither Patterson nor Zhu teaches or suggests determining if content may be encoded in a hash structure using a predefined non-lossy encoding technique. Zhu discloses that data segments may be stored in compressed (encoded) form. Ex. 1003, 5:1–5. Yet, in this embodiment (*see id.*, Fig. 3), the segment data is compressed and stored in its associated content segment (data store), but not in a hash structure. Likewise, Terao does not teach or suggest encoding data in a hash structure. Terao also does not teach or suggest comparing an encoded value to determine if it would fit within the field for containing a hash signature.

Petitioner contends both Patterson and Zhu disclose that a data segment itself could be used as its segment identifier, and that Zhu discloses compressing the contents of data segments. Pet. 28. Based on these disclosures, Petitioner reaches the conclusion that “[i]t would be obvious to one skilled in the art that the data segments in the Patterson embodiments [in a hash structure] could also be compressed with a non-lossy encoding technique” *Id.* Petitioner cites Dr. Shenoy’s testimony, which simply reiterates, without support, Petitioner’s conclusion that it would have been obvious to store data segments in Patterson’s embodiments in the compressed form as described in Zhu to make the deduplicated storage system more efficient. Shenoy Decl. ¶ 77. Neither Patent Owner nor

Dr. Shenoy accounts for the difference between storing a data segment itself in a hash structure and storing a portion of the data in encoded form directly in a hash structure, as required by claim 1. The claimed invention (*see* step 1F) applies encodings to received portions of the data and determines if the encodings would fit within a hash structure. Yet, Petitioner has not cited in the references any teaching of storing an encoding in a hash structure and Petitioner fails to explain persuasively how the combination of the three references resolves that deficiency. Instead, Petitioner and Dr. Shenoy provide mere conclusory statements to support the assertion that the combination of references teaches determining if received content can be encoded in the hash structure. *See* Pet. 28, 29.

We agree with Patent Owner that “[n]either Petitioner nor its expert explain why a [person of ordinary skill in the art] would try to encode data and then store it specifically in a hash structure” as required in step 1F. Prelim. Resp. 39. Petitioner does not adequately explain why or how a person of ordinary skill in the art would arrive at the modifications necessary to meet the requirements of step 1F. Petitioner’s reasons for combining the references – such as reducing the amount of data stored in each field or making the storage system more efficient (Pet. 27–28) – do not adequately address the modifications needed by each reference to reach the determining step (1F). *See Plantronics, Inc. v. Aliph, Inc.*, 724 F.3d 1343, 1354 (Fed. Cir. 2013) (“Where, as here, the necessary reasoning is absent, we cannot simply assume that an ordinary artisan would be awakened to modify prior art in such a way as to lead to an obviousness rejection.” (internal quotation marks and citation omitted)).

Based on the record before us, Petitioner has also not persuasively established that it would have been obvious to “mark[] the new hash structure to indicate that the field contains encoded content for the deduplicated image” as also required by claim 1 (step 1G). Petitioner’s argument related to this limitation is not supported by persuasive evidence. Instead, Petitioner presents a conditional argument “that *if* a unique identifier for a particular data segment in the reconstruction list or the ID sequence is the encoded data segment itself, there must be a way for the system to distinguish such entries” Pet. 30 (emphasis added). As explained above, however, Petitioner’s showing of an encoded data segment as a unique identifier in a hash structure is conclusory and not supported by persuasive evidence.

In support of its contention that marking the hash structure would have been obvious, Petitioner again relies on conclusory testimony from Dr. Shenoy. Pet. 31 (citing Shenoy Decl. ¶ 81). This testimony concludes, without support, that it would have been obvious to include “an indicator as to whether the unique identifier is the encoded data segment itself or a value derived from a hash function of the data segment to specify in the reconstruction process whether or not an additional lookup is required to retrieve the stored data segment.” Shenoy Decl. ¶ 81. Such conclusory testimony is entitled to little weight. *See* 37 C.F.R. § 42.65(a). Petitioner does not cite to any evidence where a hash structure is marked to indicate a field contains encoded content as claimed. Further, Petitioner has not persuasively identified a reason or need in Patterson or Zhu to differentiate a hash structure that stores an encoding as claimed.

Terao also does not teach or suggest marking the new hash structure to indicate that the field contains encoded content. According to claim 1, marking is done “so that the received content is not stored since the received content can be derived from the encoding.” Ex. 1001, 34:32–34. Yet, as Patent Owner points out, “Terao teaches that the received content is *always* stored.” Prelim Resp. 45; *see also* Terao ¶¶ 38, 39 (storing either a “data fragment per se as a characteristic value” or a “data fragment”). Thus, we do not agree that Terao teaches marking or supports modifying Patterson and Zhu to include marking as claimed.

In sum, there are multiple instances in which Petitioner has failed to show a teaching of a claim limitation in the prior art. Petitioner has attempted to fill those gaps with conclusory expert testimony that, itself, does not cite to evidentiary support. Viewing Petitioner’s evidence and argument as a whole, we are persuaded that Petitioner’s proposed combination of Patterson, Zhu, and Terao is not driven by the teachings of those references but, rather, is built on impermissible hindsight. *See In re NTP, Inc.*, 654 F.3d 1279, 1299 (Fed. Cir. 2011) (“Care must be taken to avoid hindsight reconstruction by using the patent in suit as a guide through the maze of prior art references, combining the right references in the right way so as to achieve the result of the claims in suit.” (internal quotation marks and citations omitted)).

On the record before us, we do not agree that Patterson, Zhu, and Terao, teach or suggest “determining if the received content may be encoded using a predefined non-lossy encoding technique and in which the encoded value would fit within the field for containing a hash signature” (1F) or “marking the new hash structure to indicate that the field contains encoded

content for the deduplicated image” (1G). Ex. 1001, 34:24–34. We therefore determine that Petitioner has not demonstrated a reasonable likelihood of prevailing on its challenge to claim 1 of the ’944 patent as obvious.

5. Claims 3 and 4

Claims 3 and 4 depend from claim 1. For the same reasons set forth for claim 1, Petitioner has not demonstrated a reasonable likelihood of prevailing on its challenge to claims 3 and 4 of the ’944 patent as obvious.

C. Obviousness of Claim 2 Over Patterson, Zhu, Terao, and Li

Claim 2 depends from claim 1. Petitioner does not allege Li teaches or suggests the requirements of steps 1F and 1G discussed above for claim 1 and missing from Patterson, Zhu, and Terao. *See* Pet. 36 (Li teaches “the non-lossy encoding disclosed in Zhu can be a run-length encoding”). Therefore, for the reasons set forth above, we determine that Petitioner has not demonstrated a reasonable likelihood of prevailing on its challenge to claim 2 of the ’944 patent as obvious.

III. CONCLUSION

The information presented in the Petition does not establish a reasonable likelihood that Petitioner would prevail on any of the following challenges to the patentability of the ’944 patent:

- A. Obviousness of claims 1, 3, and 4 over Patterson, Zhu, and Terao;
and
- B. Obviousness of claim 2 over Patterson, Zhu, Terao, and Li.

IV. ORDER

It is, therefore,

ORDERED that the Petition is denied as to all challenged claims and no trial is instituted.

IPR2015-01678
Patent 8,299,944 B2

PETITIONER:

David Hadden
Saina Shamilov
Fenwick & West LLP
dhadden-ptab@fenwick.com
sshamilov-ptab@fenwick.com

PATENT OWNER:

Robert Steinberg
Jonathan Link
Latham & Watkins LLP
bob.steinberg@lw.com
jonathan.link@lw.com
ACTIFIOINC.IPR ACTIONS.LWTEAM@lw.com