

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE, INC.,
Petitioner,

v.

REALTIME DATA LLC,
Patent Owner.

Case IPR2016-01739
Patent 8,880,862 B2

Before DEBRA K. STEPHENS, GEORGIANNA W. BRADEN, and
JASON J. CHUNG, *Administrative Patent Judges*.

BRADEN, *Administrative Patent Judge*.

FINAL WRITTEN DECISION
35 U.S.C. § 318 and 37 C.F.R. § 42.73

I. INTRODUCTION

We have jurisdiction to hear this *inter partes* review under 35 U.S.C. § 6, and this Final Written Decision is issued pursuant to 35 U.S.C. § 318(a) and 37 C.F.R. § 42.73. For the reasons that follow, we determine that Petitioner has shown by a preponderance of the evidence that claims 5, 35–46, 97, 98, and 112 (“the challenged claims”) of U.S. Patent No. 8,880,862 B2 (Ex. 1001, “the ’862 Patent”) are unpatentable.

A. Procedural History

Apple, Inc. (“Petitioner”) filed a Petition (Paper 2, “Pet.”) requesting an *inter partes* review of the challenged claims of the ’862 Patent. Realtime Data, LLC (“Patent Owner”) timely filed a Preliminary Response (Paper 6, “Prelim. Resp.”).

Pursuant to 35 U.S.C. § 314(a), we instituted an *inter partes* review of (1) Claims 5, 35–46, and 97 as unpatentable under 35 U.S.C. § 103(a)¹ in view of Settsu²; (2) Claims 5, 35–46, 97, 98, and 112 as unpatentable under 35 U.S.C. § 103(a) in view of Settsu and Zwiegincew³; (3) Claims 5, 35–46, and 97 as unpatentable under 35 U.S.C. § 103(a) in view of Settsu and Dye⁴; and (4) Claims 5, 35–46, 97, 98, and 112 as unpatentable under 35 U.S.C.

¹ The Leahy-Smith America Invents Act (“AIA”) included revisions to 35 U.S.C. § 100 et seq. effective on March 16, 2013. The ’862 patent issued from an application filed before March 16, 2013; therefore, we apply the pre-AIA versions of the statutory bases for unpatentability.

² U.S. Patent No. 6,374,353 B1, filed Mar. 3, 1999, issued Apr. 16, 2002 (Ex. 1006, “Settsu”).

³ U.S. Patent No. 6,317,818 B1, filed Mar. 30, 1999, issued Nov. 13, 2001 (Ex. 1010, “Zwiegincew”).

⁴ U.S. Patent No. 6,145,069, filed Apr. 26, 1999, issued Nov. 7, 2000 (Ex. 1008, “Dye”).

§ 103(a) in view of Settsu, Zwiegincew, and Dye. See Paper 7 (“Dec. to Inst.”), 30.

After institution of trial, Patent Owner filed a Patent Owner Response (Paper 15, “PO Resp.”), to which Petitioner filed a Reply (Paper 17, “Reply”). Patent Owner also filed Objections to Evidence in Petitioner’s Reply (Paper 18) and a Motion to Exclude Evidence (Paper 25). Petitioner opposed the Motion to Exclude (Paper 26) and Patent Owner submitted a Response in support of its Motion to Exclude (Paper 29). In addition, Patent Owner filed a list of alleged improper reply arguments (Paper 19) to which Petitioner filed a Reply (Paper 20).

An oral argument was held on January 8, 2018. A transcript of the oral argument is included in the record. Paper 30 (“Tr.”).

B. Related Proceedings

The parties identify the following cases as related to the challenged patent: *Realtime Data, LLC v. Microsoft Corporation*, Case No. 4:14-cv-00827 (E.D. Tex.), *Realtime Data, LLC v. Microsoft Corporation*, Case No. 6:15-cv-00885 (E.D. Tex.), and *Realtime Data, LLC v. Apple, Inc.*, Case No. 3:16-cv-02595 (N.D. Cal.) (transferred from *Realtime Data, LLC v. Apple, Inc.*, Case No. 6:15-cv-00885 (E.D. Tex.)). Pet. 1; Paper 5, 2.

C. The ’862 Patent

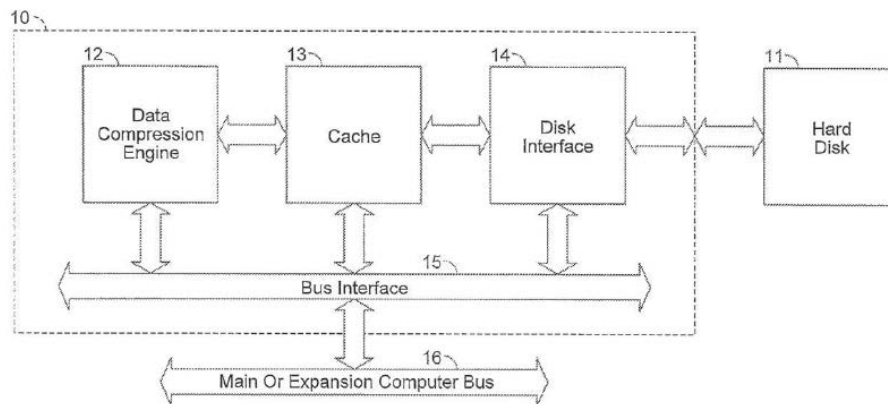
The ’862 Patent relates to “providing accelerated loading of operating system and application programs upon system boot or application launch,” and the use of data compression and decompression techniques for such purpose. Ex. 1001, 1:20–26. The specification discusses the limits of prior art storage devices, particularly the significant bandwidth limitations of

“mass storage devices” such as hard disk drives. *Id.* at 1:43–57, 2:9–18.

According to the specification,

“[A]ccelerated” data storage comprises receiving a digital data stream at a data transmission rate which is greater than the data storage rate of a target storage device, compressing the input stream at a compression rate that increases the effective data storage rate of the target storage device and storing the compressed data in the target storage device.

Id. at 5:41–47. One embodiment of the ’862 Patent is illustrated in Figure 1, reproduced below.



As shown in Figure 1, data storage controller 10 is “operatively connected” to hard disk 11 and to host system’s bus 16. *Id.* at 5:63–6:53. Controller 10 includes cache 13 for data storage/preloading, and data compression engine 12 for data compression/decompression. *Id.* at 5:63–6:53, 20:50–22:11.

The ’862 Patent explains that, following reset or power on of a computer system, the “initial bus commands inevitably instruct the boot device controller [e.g., controller 10] to retrieve data from the boot device (such as a disk) [e.g., hard disk 11] for the operating system.” *Id.* at 20:36–49.

D. Illustrative Claims

As noted above, an *inter partes* review was instituted as to claims 5, 35–46, 97, 98, and 112 of the ’862 Patent. Dec. to Inst. 30. Of the

challenged claims, claim 5 is the only independent claim. Claim 5 is illustrative of the challenged claims, and is reproduced below:

5. A method for booting a computer system, the method comprising:
 - storing boot data in a compressed form that is associated with a portion of a boot data list in a first memory;
 - loading the stored compressed boot data from the first memory;
 - accessing the loaded compressed boot data;
 - decompressing the accessed compressed boot data;
 - utilizing the decompressed boot data to at least partially boot the computer system; and
 - updating the boot data list,wherein the loading, the accessing, and the decompressing occur within a period of time which is less than a time to access the boot data from the first memory if the boot data was stored in the first memory in an uncompressed form.

Ex. 1001, 26:60–27:8.

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. 37 C.F.R. § 42.100(b); see *Cuozzo Speed Techs., LLC v. Lee*, 136 S. Ct. 2131, 2144–46 (2016) (“We conclude that the regulation represents a reasonable exercise of the rulemaking authority that Congress delegated to the Patent Office.”). Under that standard, and absent any special definitions, we give claim terms their ordinary and customary meaning, as would be understood by one of ordinary skill in the art at the time of the invention. *In re Translogic Tech., Inc.*, 504 F.3d 1249, 1257 (Fed. Cir. 2007). An inventor, however, may provide a meaning for a term that is different from its ordinary meaning by defining

the term in the specification with “reasonable clarity, deliberateness, and precision.” *In re Paulsen*, 30 F.3d 1475, 1480 (Fed. Cir. 1994).

Limitations, however, are not to be read from the specification into the claims. *In re Van Geuns*, 988 F.2d 1181, 1184 (Fed. Cir. 1993). In addition, the Board may not “construe claims during [an *inter partes* review] so broadly that its constructions are *unreasonable* under general claim construction principles.” *Microsoft Corp. v. Proxyconn, Inc.*, 789 F.3d 1292, 1298 (Fed. Cir. 2015).

The parties dispute the proper construction of the terms “boot data list” and “non-accessed boot data.”

1. “*boot data list*”

Patent Owner contends the term “boot data list” should mean “record used to identify and load boot data into memory.” PO Resp. 15. According to Patent Owner, both the ’862 patent’s specification and the provisional application to which the ’862 patent claims priority establish that the claimed “boot data list” is a record of boot data separate from the boot data itself. *Id.* at 15–16. Patent Owner argues that “[b]oot data comprises information such as program code relating to portions of the operating system and certain application programs.” *Id.* (citing Ex. 1001, Abs., 3:48–50; Ex. 2010, 58). Patent Owner reasons that the system stores boot data in a compressed form on a boot device (*id.* at 16 (citing Ex. 1001, Abs., 3:51–52, 3:60–61; Ex. 2010, 58)), loads boot data into memory upon initialization of the computer system (*id.* (citing Ex. 1001, Abs., 3:45–46, 4:16–17; Ex. 2010, 58)), and services requests for boot data using the loaded boot data (*id.* (citing Ex. 1001, Abs., 3:46–47, 4:1–3, 4:17–19, 21:45–59, Fig. 7B)).

Patent Owner then argues that “the intrinsic evidence describes a ‘boot data list’ as comprising a list of data—specifically, boot data—that is to be used

for booting a computer system.” *Id.* (citing Ex. 1001, 3:44–45, 4:15–16; Ex. 2010, 58). Patent Owner further argues that “[i]n one exemplary embodiment, a data storage controller retrieves and reads the ‘boot data list’ upon power-on/reset and preloads the boot data specified on the list into memory.” *Id.* at 17 (citing Ex. 1001, 21:43–48, Figs. 7B, 8A, 8B).

Petitioner contests Patent Owner’s proffered construction arguing that it is overly narrow and improperly imports limitations. Reply 1. Petitioner contends boot data list should be given its ordinary meaning and at least be construed broadly enough to include a list of data associated with data requests expected to result from a system power-on/reset. *Id.* at 2 (citing Pet. 13–16, 19–30). According to Petitioner, Patent Owner improperly attempts to import functional use to the term “boot data list,” which renders other claim language directed to use of the boot data list redundant and unnecessary. *Id.* Specifically, Petitioner argues that Patent Owner’s reliance on dependent claim 6 is misplaced, because claim 6 explicitly defines the “boot data list” as being “**used for booting the system,**” not “used to identify and load boot data into memory.” *Id.* at 2–3 (citing PO Resp. 17). Petitioner also argues that the terms “load” and “into memory” are additional features added in claim 6 that would be rendered duplicative and redundant if also imported into the term “boot data list.” *Id.* at 3. Petitioner additionally notes that the term “identify” is not used in the claims (or elsewhere in the specification). Petitioner concludes that the claims do not support Patent Owner’s attempt to limit “boot data list” as being “used to identify and load boot data into memory.” *Id.*

Petitioner further contends that Patent Owner’s proposed claim construction is not supported by the ’862 patent’s specification, specifically the description of Figure 7B. *Id.* at 3 (citing PO Resp. 17–20). According to

Petitioner, Patent Owner recognizes that this description is merely “one exemplary embodiment” of the ’862 patent, but fails to explain sufficiently why the claimed boot data list should be limited to this embodiment. *Id.* (citing PO Resp. 17). Additionally, Petitioner argues that the ’862 patent does not limit use of the boot data list “to identify and load boot data into memory,” as Patent owner contends, citing to the ’862 patent’s disclosure of “a list of boot data ***used for booting a computer system.***” *Id.* at 2 (citing Ex. 1001, Abstract, 3:42–59).

We are charged with interpreting claim terms according to their broadest reasonable construction in light of the specification of the patent in which they appear. 37 C.F.R. § 42.100(b). Therefore, we consult the patent’s specification to help clarify the meaning of claim terms. *Trading Techs. Int’l, Inc. v. eSpeed, Inc.*, 595 F.3d 1340, 1352 (Fed. Cir. 2010) (holding claims “must be read in view of the specification, of which they are a part” (quoting *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 979 (Fed. Cir. 1995) (en banc), *aff’d*, 517 U.S. 370 (1996))). We must be careful, however, not to import improperly limitations into the claims or to read a particular embodiment appearing in the written description into the claim if the claim language is broader than the embodiment. *In re Van Geuns*, 988 F.2d 1181, 1184 (Fed. Cir. 1993). Given our mandate under 37 C.F.R. § 42.100(b) and the patent’s various descriptions of “boot data list,” we understand that the examples laid out in the specification are exemplary and are not to be read as limitations in the claims.

Patent Owner’s proffered claim construction imports selected limitations from specific embodiments in the specification into the claim and provides an overly narrow interpretation of the claim term. Additionally, Patent Owner has not pointed to any definitions or disavowals in the

specification or otherwise clearly explained why the specification's disclosure of "boot data lists" should not guide our claim construction analysis. Thus, we are unpersuaded by Patent Owner's contentions, which appear to be based on a narrow reading of the claim that is inconsistent with the broadest reasonable interpretation of the claims.

Accordingly, we decline to adopt Patent Owner's claim construction as it would limit unnecessarily the scope of the claims. Instead, we find the broadest reasonable construction of "boot data list" to encompass a "list of boot data."

2. "*non-accessed boot data*"

Patent Owner contends that the term "non-accessed boot data," as used in claims 96, 100, 102, and 106, should mean "boot data identified in the boot data list that was not requested during system boot-up." PO Resp. 20 (citing Ex. 2008 ¶¶ 55–60, 61–66). According to Patent Owner, "[t]he specification explains that 'non-accessed boot data' is boot data that has been retrieved and recorded in the boot data list during a previous system boot-up but was not requested during a subsequent system boot-up." *Id.* at 20–21. Patent Owner argues that if the boot data is not requested during system boot-up, then that boot data is "excluded" from the boot data list. *Id.* at 21 (citing Ex. 1001, Fig. 7B, 22:5–11).

Petitioner contests Patent Owner's position arguing that the intrinsic record does not limit functionally the term "non-accessed" to "not requested" or limit temporally the term "non-accessed" to "during system boot-up." Reply 5. Rather, according to Petitioner, under the broadest reasonable interpretation, a person of ordinary skill in the art would have viewed the term "non-accessed boot data" per its ordinary meaning as simply boot data that was not accessed. *Id.* (citing Ex. 1003 ¶¶ 203–205).

Petitioner argues that the embodiments in the specification are non-limiting examples. *Id.* at 5–6. Petitioner further argues that Patent Owner’s reliance on the embodiment of Figure 7B is misplaced because the very next embodiment illustrated in Figures 8a and 8b includes references to a “non-requested data block” “[d]uring the **application launch process**.” *Id.* at 6 (citing Ex. 1001, 22:12–23:26; Ex. 1002 (Part 1), 156–157, 160–162).

Petitioner notes that Patent Owner’s citations include the application launch embodiment, which contradicts limiting non-accessed boot data to only data “not requested during system boot-up.” *Id.* Petitioner concludes that adopting Patent Owner’s construction would exclude improperly a specific embodiment (during application launch) that Patent Owner cites now in support of its construction and also cited during prosecution to show written description support for the dispute term. *Id.*

We determine that Patent Owner’s proposed construction is unduly narrow and improperly attempts to import limitations from the specification. Rather, we agree with Petitioner’s position and we construe “non-accessed boot data” as “boot data that has not been accessed.”

3. Additional claim terms

We determine that no additional claim terms require express construction at this stage (*see Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999) (Only terms which are in controversy need to be construed, and only to the extent necessary to resolve the controversy)).

B. Principles of Law

A claim is unpatentable under 35 U.S.C. § 103(a) if “the differences between the subject matter sought to be patented 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 skill in the art; and (4) objective evidence of nonobviousness, i.e., secondary considerations. *See Graham v. John Deere Co. of Kansas City*, 383 U.S. 1, 17–18 (1966).

“A determination of whether a patent claim is invalid as obvious under § 103 requires consideration of all four *Graham* factors, and it is error to reach a conclusion of obviousness until all those factors are considered.” *Apple v. Samsung Elecs. Co., Ltd.*, 839 F.3d 1034, 1048 (Fed. Cir. 2016) (en banc) (citations omitted). “This requirement is in recognition of the fact that each of the *Graham* factors helps inform the ultimate obviousness determination.” *Id.*

“In an [*inter partes* review], the petitioner has the burden from the onset to show with particularity why the patent it challenges is unpatentable.” *Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1363 (Fed. Cir. 2016) (citing 35 U.S.C. § 312(a)(3) (requiring *inter partes* review petitions to identify “with particularity . . . the evidence that supports the grounds for the challenge to each claim”)). This burden never shifts to Patent Owner. *See Dynamic Drinkware, LLC v. Nat’l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015) (citing *Tech. Licensing Corp. v. Videotek, Inc.*, 545 F.3d 1316, 1326–27 (Fed. Cir. 2008)) (discussing the burden of proof in *inter partes* review). Furthermore, Petitioner cannot satisfy its

burden of proving obviousness by employing “mere conclusory statements.” *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016).

Thus, to prevail in an *inter partes* review, Petitioner must explain how the proposed combinations of prior art would have rendered the challenged claims unpatentable. At this final stage, we determine whether a preponderance of the evidence of record shows that the challenged claims would have been obvious over the proposed combinations of prior art.

We analyze the instituted grounds of unpatentability in accordance with the above-stated principles.

C. Level of Ordinary Skill in the Art

In determining whether an invention would have been obvious at the time it was made, we consider the level of ordinary skill in the pertinent art at the time of the invention. *Graham*, 383 U.S. at 17. “The importance of resolving the level of ordinary skill in the art lies in the necessity of maintaining objectivity in the obviousness inquiry.” *Ryko Mfg. Co. v. Nu-Star, Inc.*, 950 F.2d 714, 718 (Fed. Cir. 1991). The person of ordinary skill in the art is a hypothetical person who is presumed to have known the relevant art at the time of the invention. *In re GPAC, Inc.*, 57 F.3d 1573, 1579 (Fed. Cir. 1995). The level of ordinary skill in the art may be reflected by the prior art of record. *Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001). Factors that may be considered in determining the level of ordinary skill in the art include, but are not limited to, the types of problems encountered in the art, the sophistication of the technology, and educational level of active workers in the field. *GPAC*, 57 F.3d at 1579. In a given case, one or more factors may predominate. *Id.* Generally, it is easier to establish obviousness under a higher level of ordinary skill in the art. *Innovation*

Toys, LLC v. MGA Entm't, Inc., 637 F.3d 1314, 1323 (Fed. Cir. 2011) (“A less sophisticated level of skill generally favors a determination of nonobviousness . . . while a higher level of skill favors the reverse.”).

Petitioner’s declarant, Charles J. Neuhauser, Ph.D. (“Dr. Neuhauser”), opines that a person of ordinary skill in the art relevant to the ’685 patent would have had “a Bachelor’s Degree in electrical engineering, computer engineering, or a related area of study” as well as “between three and five years of practical experience in the design and implementation of computer systems, such as personal computers.” Ex. 1003 ¶ 15. Dr. Neuhauser further opines that, in the alternative, a person with “a Master’s Degree in the area of electrical engineering, computer engineering, or a related area of study and somewhat less practical experience would be similarly qualified.” *Id.*

Patent Owner does not dispute Dr. Neuhauser’s testimony. *See generally* PO Resp. Patent Owner’s Declarant, Dr. Godmar Back (“Dr. Back”), however, provides his own assessment regarding a person of ordinary skill in the art relevant to the ’862 patent and agrees with Dr. Neuhauser’s testimony. Ex. 2008 ¶ 50.

We do not observe a meaningful differences between the parties’ assessments of a person of ordinary skill in the art. We further note that either assessment appears consistent with the level of ordinary skill in the art at the time of the invention as reflected in the prior art in the instant proceeding. *See Okajima v. Bourdeau*, 261 F.3d 1350, 1355 (Fed. Cir. 2001). Our analysis in this Decision is supported by either assessment. Based on our review of the ’862 patent, the types of problems and solutions described in the ’862 patent and cited prior art, and the testimony of Dr. Neuhauser and Dr. Back, we adopt and apply Dr. Back’s definition of a

person of ordinary skill in the art at the time of the claimed invention for purposes of this Decision.

D. Overview of the Asserted Prior Art

1. Settsu

Settsu is a U.S. Patent titled “Information Processing Apparatus Method [sic] of Booting Information Processing Apparatus at a High Speed” and relates to “[a] method of booting up an information processing apparatus. . . .” Ex. 1006, [54], [57]. One embodiment of Settsu is shown in Figure 1, reproduced below.

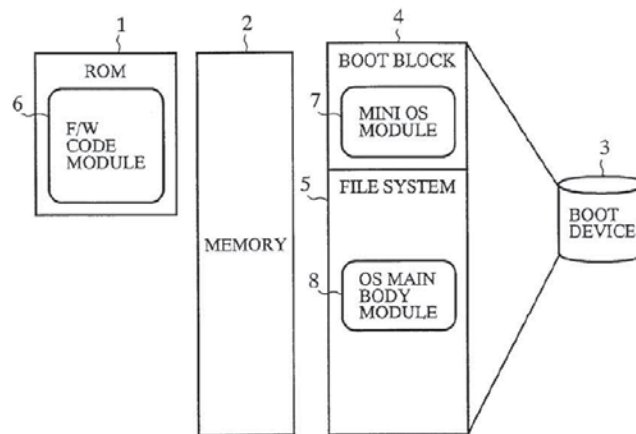
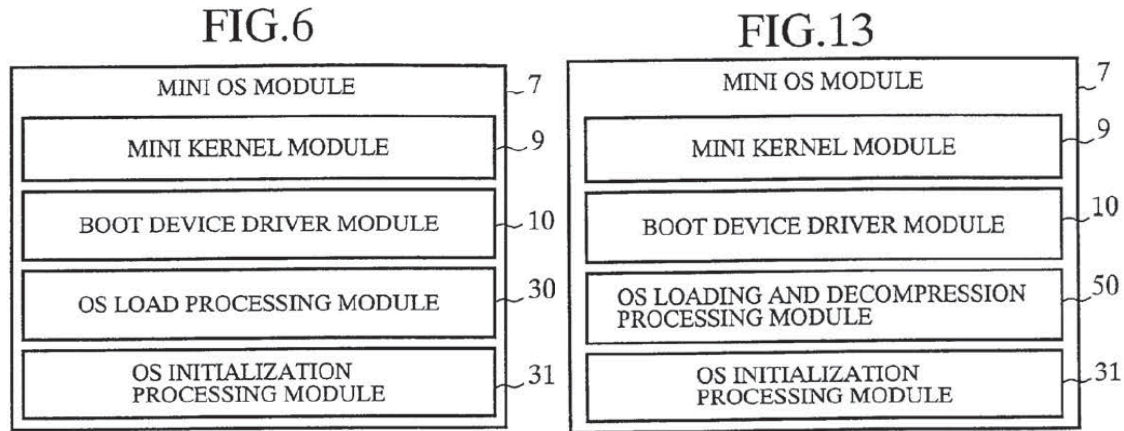


Figure 1 illustrates an information processing apparatus with boot device 3, firmware 6, memory device 2, and boot block 4. *Id.* at 8:2–8. According to Settsu, boot block 4 includes a mini operating system (OS) module that has OS functions required for bootstrap processing, and an OS main body module located in file system 5 that provides the apparatus with the OS functions not included on mini OS module 7. *Id.* at 8:13–21.

Two other embodiments of Settsu are shown in Figures 6 and 13, reproduced below.



Figures 6 and 13 illustrate dividing up Mini OS Module 7. *Id.* at 10:59–11:3, 13:66–14:6. Settsu discloses that Mini OS Module 7 can include OS loading and decompression processing module 50 so that modules can be stored as compressed files on a boot device. *Id.* at 14:6–12, 14:58–63. Each of these modules is decompressed each time it is loaded into memory, and “the time required for I/O [(input/output)] processing can be reduced” as a result, which “provides an advantage of being able to further reduce the time required for booting up the information processing apparatus.” *Id.* at 14:64–15:4.

2. Dye

Dye is a U.S. Patent titled “Parallel Decompression and Compression System and Method for Improving Storage Density and Access Speed for Non-Volatile Memory and Embedded Memory Devices.” Ex. 1008, at [54]. Dye relates to controllers for flash or embedded memory that include data compression and decompression engines “for increased effective memory density and improved bandwidth.” *Id.* at 1:17–22, 2:42–46. According to Dye, such a controller enables conventional flash memory to “achieve higher bandwidth, more effective density, with less system power and noise. . . .” *Id.* at 3:3–12, 3:23–28. The technology permits data to be “saved in either a normal or compressed format, retrieved from the Flash Memory Array for

MPU [(microprocessing unit)] execution in a normal or compressed format, or transmitted and stored on a medium in a normal or compressed format.”

Id. at 3:66–4:8.

Figure 10B of Dye is reproduced below.

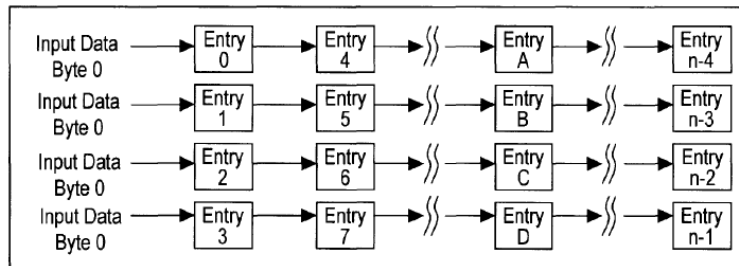


Fig. 10B
(New Art)

Figure 10B, above, illustrates a preferred embodiment “provid[ing] a parallel implementation of dictionary based (or history table based) compression/decompression.” Ex. 1008 18:61–63. In the preferred embodiment illustrated in Dye’s Figure 10B, the history table becomes a four symbol parallel flow. *Id.* at 19:15–17. “[Four] symbols are analyzed in parallel, and multiple compressed outputs may also be provided in parallel.” *Id.* at 19:18–19. “Other alternate embodiments may contain a plurality of compression windows for decompression of multiple streams. . . .” *Id.* at 19:19–21.

Figure 13 of Dye is reproduced below.

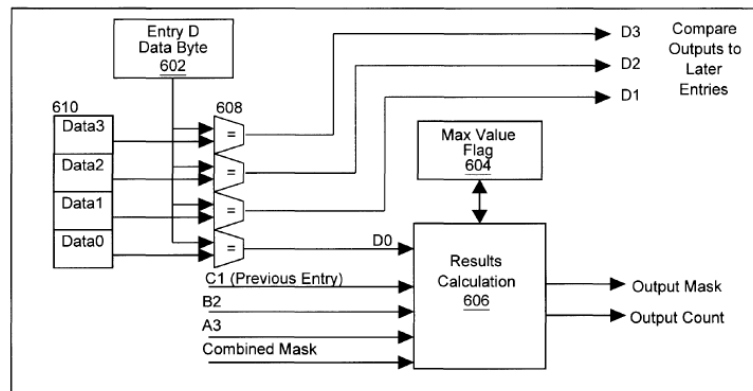


Fig. 13

Figure 13, above, is a hardware diagram illustrating “operation of the parallel compression algorithm.” *Id.* at 22:66–67. Each entry of the history table contains a symbol of data, which is compared with the input stream. *Id.* at 23:1–3. In Figure 13, Entry D Data Byte 602 is compared with each symbol of input data stream 610, shown as four data bytes, Data 0, 1, 2, and 3. *Id.* at 23:5–7. Comparators 608 compare each data byte to Entry D Data Byte 602, generating four compare signals (D0 through D3), with D1 through D3 being used by the next entry in the history table and D0 being used by Results Calculation 606. *Id.* at 23:7–55. Results Calculation 606 sends Output Mask and Output Count to logic shown in Dye’s Figure 14. *Id.* at 23:19–24.

3. *Zwiegincew*

Zwiegincew is a U.S. Patent titled “Pre-Fetching of Pages Prior to a Hard Page Fault Sequence,” and it recognizes problems of slow boot that result when hard page faults occur during the boot process. Ex. 1010, [54], 1:45–51, 2:12–15, 6:50–51. To improve boot speed, *Zwiegincew* proposes pre-fetching, from a hard disk to memory, pages that are expected to be

requested during the boot process, thereby reducing occurrence of hard page faults. *Id.* at [57], 1:5–3:55. “Copies of, or references to, the . . . pages are stored in a scenario file” and, “[w]hen a hard page fault scenario is detected, a corresponding scenario file is fetched from disk storage and the determined pages, or copies thereof, are transferred into RAM.” *Id.* at [57].

Zwiegincew also recognizes benefits of compressing pre-fetched page data. For instance, Zwiegincew’s system includes “a disk compressor/decompressor,” which employs “compression algorithms” on pre-fetched data to achieve pre-fetch time improvements. *Id.* at 8:66–9:13, Figs. 1–2.

E. Alleged Obviousness of the Challenged Claims in View of Settsu

Petitioner contends Settsu teaches or suggests each element of claims 5, 35–46, and 97. Pet. 16–60. Patent Owner disputes Petitioner’s contentions. PO Resp. 24–31. For reasons that follow, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that the challenged claims of the ’862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu.

1. *Analysis of Cited Art as Applied to Independent Claim 5*

Petitioner contends the teachings of Settsu renders every element of claim 5 of the ’862 patent obvious. Pet. 16–38. Patent Owner disputes Petitioner’s contentions. PO Resp. 24–31. Patent Owner specifically argues that Petitioner fails to show that Settsu teaches “updating the boot data list.” *Id.* at 24–30.

After considering the parties’ arguments and evidence, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that challenged claim 5 of the ’862 patent is

unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu. We address the issues disputed by Patent Owner in more detail below.

Petitioner contends Settsu describes a boot data list in at least two ways: “(1) in the form of lists of boot data that are stored within the OS functional module files themselves; and (2) in the form of lists of boot data that are referenced by mini OS module 7 in the course of booting the OS.” Pet. 19–21, 34 (citing Ex. 1003 ¶¶ 70, 77, 116; Ex. 1006, 5:39–51, 7:65–8:23, 13:49–65, 14:44–52, 16:7–17:62, Figs. 12, 20 (ST213)). With respect to the function definition file, Petitioner notes that “Settsu describes that the function definition file includes a ‘listing [of] a minimum number of OS functional modules . . . required for the application module 70 to run on the OS.’” *Id.* at 26 (citing Ex. 1006, 16:26–31). Petitioner further notes that “Settsu’s processing module 72 refers to the function definition file 71 and ‘loads one of the plurality of OS functional modules listed in the function definition file 71 into the memory 2.’” *Id.* (citing Ex. 1006, 16:65–17:1).

Petitioner then argues that Settsu’s description renders obvious updating both types of lists. *Id.* at 28 (citing Ex. 1003 ¶ 117). According to Petitioner, Settsu OS main body module is divided into “a plurality of functional modules” that “are separately stored as compressed files” in “file system 5 of boot device 3.” *Id.* at 34 (citing Ex. 1006, 13:55–65). From this description, Petitioner argues that a person of ordinary skill in the art would have understood that OS modules, such as Settsu’s, require updates when new, updated code becomes available. *Id.*

Petitioner supports its position with the Declaration of Dr. Neuhauser. *Id.* at 35 (citing Ex. 1003 ¶ 117). Dr. Neuhauser specifically opines that an OS functional module file stored on boot device 3, as described by Settsu, includes a list of data necessary for starting the OS and, thus, is a boot data

“list” (that is, a list of data) under a broadest reasonable interpretation. Ex. 1003 ¶ 78 (citing Ex. 1014 for a definition of the term “file” as “[a] complete, named collection of information, such as a program, a set of data used by a program, or a user-created document” that “binds a conglomeration of instructions . . . into a coherent unit”). As supported by the cited definition, Dr. Neuhauser testifies one of ordinary skill would have understood Settsu’s compressed OS functional module files to be lists of data (including, e.g., program instructions). *Id.* Based on Dr. Neuhauser’s testimony, Petitioner concludes that through its description of storing compressed OS functional module files on boot device 3, Settsu teaches storing boot data in a compressed form that is associated with a portion of a boot data list as recited by claim 5. Pet. 22 (citing Ex. 1003 ¶ 79).

Dr. Neuhauser further testifies that to benefit from updates in Settsu’s system or simply to account for changes in user preference, a person of ordinary skill in the art would have understood that Settsu’s system updates the OS data stored in boot device 3. Ex. 1003 ¶ 117 (citing Ex. 1026, 110–112) (describing regular updates to the Windows 98 operating system). According to Dr. Neuhauser, to accomplish these updates, a person of ordinary skill in the art would have understood that updates occur to boot data lists in Settsu; namely, updates to lists of boot data included within OS functional module files stored in boot device 3, as well as updates to lists of boot data, such as function definition file 71 or OS main body module 8. *Id.* (citing Ex. 1006, 5:39–51, 13:55–65, 14:44–52, 16:7–17:62, Figs. 12, 20 (ST213)).

Patent Owner contests Petitioner’s position. First, Patent Owner contends that Petitioner fails to show Settsu discloses a “boot data list” and incorrectly relies on Settsu inherently teaching the limitation. PO Resp. 24–

25. Specifically, Patent Owner argues that Petitioner incorrectly presumes that “each electronic file includes a list of data stored within the file” and that Petitioner relies on an incorrect definition of “file” to support its position. *Id.* at 26 (citing Pet. 34–35). Second, Patent Owner contends that a person of ordinary skill in the art would have understood that most, if not all, of the operating system update would not change the list of OS modules in Settsu’s system. *Id.* at 29–30. Lastly, Patent Owner contends the Examiner expressly considered Settsu during prosecution and allowed the claims in view of Settsu’s teachings once the claims were amended to include “updating the boot data list” limitation. *Id.* at 30 (citing Ex. 1002 (Part 1), 229–261).

Patent Owner’s arguments that Petitioner fails to show the asserted art teaches “boot data list” (PO Resp. 24–25) are unpersuasive because Settsu explicitly teaches function definition file 71 lists OS modules that must be loaded and initialized during a boot process. *See* Ex. 1006, 16:48–17:20, Fig. 20. As shown in Settsu’s Figure 20, reproduced below, AP execution and OS load processing module 72 in step ST213, loads one of the plurality of OS functional modules listed in function definition file 71 into memory 2.

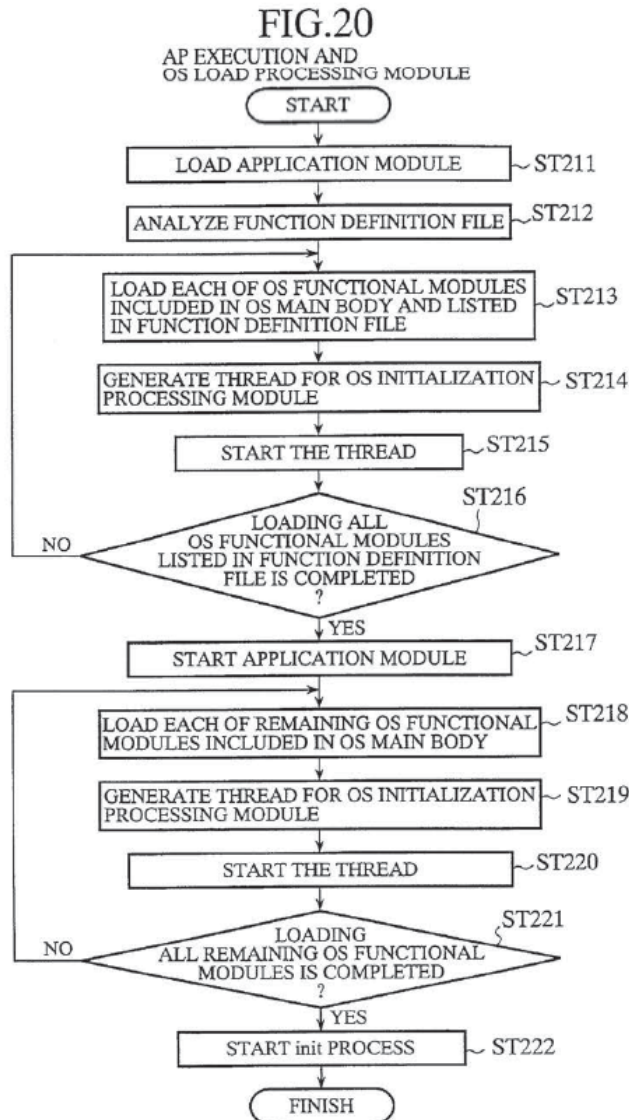


Figure 20 illustrates an AP execution and OS load processing module. Accordingly, we determine that this list constitutes a boot data list regardless of whether we apply our construction or Patent Owner’s broader construction of the term.

Additionally, we do not agree with Patent Owner’s position that Settsu fails to teach “updating a boot list.” Rather, we agree with the testimony of Dr. Neuhauser regarding the similarities between Settsu’s OS and the functionality of the Windows 98 OS, and why a person of ordinary skill in the art would have considered updates to have occurred to the boot

data lists in Settsu. *See* Ex. 1003 ¶ 117; Ex. 1003, 5:39–51, 13:55–65, 14:44–52, 16:7–17:62, Figs. 12, 20(ST213). Specifically, we determine that a person of ordinary skill in the art would have understood that OS modules, such as Settsu’s, would have required updates when new, updated code becomes available. *See* Ex. 1003 ¶ 117 (citing O’Reilly 110–112). We agree with Dr. Neuhauser that to accomplish these updates, a person of ordinary skill in the art would have understood that updates occur to boot data lists in Settsu; namely, updates to lists of boot data included within OS functional module files stored in boot device 3, as well as updates to lists of boot data, such as the function definition file 71 or the OS main body module 8. *See id.*

Given the explicit teaching of Settsu and the credible testimony of Dr. Neuhauser regarding the state of the art and the knowledge of one of skill the art, we are unmoved by Patent Owner’s argument regarding the prosecution history of the ’862 patent or that the Examiner fully understood the implications of Settsu’s Figure 20. Accordingly, based on the entirety of the record and evidence, and crediting the testimony from Dr. Neuhauser (*see* Ex. 1003 ¶¶ 78, 116–118), we determine that “updating the boot data list” is taught or suggested by Settsu, and the subject matter of the claim as whole would have been obvious to a person of ordinary skill in art in view of Settsu.

2. *Remaining Challenged Claims*

The Petition sets forth detailed contentions and supporting evidence alleging that claims 35–46 and 97 are obvious in light of the teachings of Settsu. Pet. 38–58. For instance, for the “plurality of files” limitation of claims 35, 40, and 42, Petitioner relies on Settsu’s teachings of an OS “main body module” that is “divided into a plurality of functional modules” that

“are separately stored as compressed files” in “file system 5 of boot device 3.” Pet. 38 (citing Ex. 1006, 2:5–67, 13:55–65). According to Petitioner, as supported by Dr. Neuhauser’s testimony, a person of ordinary skill in the art would have understood a program, like the operating system loaded by Settsu into memory 2, typically includes a plurality of files. *Id.* (citing Ex. 1003 ¶ 131). With respect to the claims reciting compression or decompression, such as claims 37, 38, 39, 43, and 97, Petitioner identifies teachings in Settsu that describe loading OS files from a boot device’s file system “into [a] memory 2 as compressed data,” and then decompressing the loaded OS files to speed up a computer system’s boot process. *Id.* at 42–43 (citing Ex. 1006, at [57], 1:1–4, 1:51–57, 3:6–9, 13:49–15:5). As for those claims that recite a form of dictionary encoding or the Lempel-Ziv algorithm for compression, such as claims 44 and 45, Petitioner notes that the Lempel-Ziv algorithm, which uses a form of dictionary encoding to compress data, was well-known at the time of the ’862 patent. *Id.* at 50 (citing Ex. 1003 ¶ 176). According to Petitioner, a skilled artisan would have found it obvious to use the well-known Lempel-Ziv algorithm to compress the OS files in Settsu. *Id.* at 50–51 (citing Ex. 1003 ¶¶ 175–177).

Patent Owner generally presents the same or similar arguments for claims 35–45 and 97 as presented for claim 1, which are unpersuasive on the present record for similar reasons as for claim 1. Regarding claim 46, however, Patent Owner contends that Settsu fails to teach the use of a “plurality of encoders” to “encode the compressed boot data.” PO Resp. 44. Specifically, Patent Owner argues Petitioner’s assertion that Settsu alone renders the use of a plurality of encoder, is pure attorney argument unsupported by any evidence, including expert testimony. *Id.* According to Patent Owner, Dr. Neuhauser does not offer an opinion or a reasoned

analysis that Settsu alone renders claim 46 obvious. *Id.* (citing Ex. 2011, 319:18–20 (Q: You do not provide an opinion that Claim 46 is obvious in view of Settsu alone, correct? A: That’s correct.)).

Petitioner contests Patent Owner’s position. Petitioner first acknowledges that, although “Settsu’s system stores operating system files in compressed form on boot device 3,” “Settsu fails to explicitly disclose a step of compressing the OS files.” Pet. 40 (citing Ex. 1003 ¶ 135). Yet, according to Petitioner, a person of ordinary skill in the art “would have understood that a step of compressing the OS files would have been needed to arrive at the OS files in compressed form.” *Id.* Petitioner argues that “Settsu simply omits mention of this necessary step and presupposes existence of the compressed OS files.” *Id.* Thus, Petitioner asserts that “a [person of ordinary skill in the art] would have understood and found obvious that a step of compressing the OS files occurs for Settsu’s OS files to be used in compressed form.” *Id.*

Petitioner further contends that “[t]o perform this compression, a [person of ordinary skill in the art] would have understood that a form of compression or encoding would have been used by a compressor or encoder to compress or encode the data.” Pet. 54. Petitioner relies on the testimony of Dr. Neuhauser to support its position. Dr. Neuhauser testifies that it was well-known at the time of the ’862 patent and Settsu to utilize a plurality of encoders to encode compressed data. Ex. 1003 ¶¶ 190–192 (citing Ex. 1008, 14:9–30, 8:43–19:30, 22:66–23:24, Figs. 10A, 10B, 13; Ex. 1009, 12:61–13:7, 13:52–56, 14:49–50, 29:13–14). Petitioner, thus, asserts that “from Settsu’s description of compressed OS files and the well-known solution of utilizing a plurality of encoders to encode compressed data, a [person of ordinary skill in the art] would have found it obvious to use a plurality of

encoders to compress Settsu's OS files and to provide the compressed OS files to boot device 3." Pet. 54 (citing Ex. 1003 ¶¶ 190–192).

We agree with Petitioner that given the explicit disclosure in Settsu of storing compressed files and decompressing files, a person of ordinary skill in the art would have understood that at least compression and decompression would occur. We do not agree, however, that a person of ordinary skill in the art would have found it obvious from Settsu's teachings that multiple or a plurality of encoders would have been used to achieve the compression of files. There does not appear to be any suggestion or teaching in Settsu that would lead a person of ordinary skill in the art to use a plurality of encoders. Although we credit the testimony of Dr. Neuhauser that using multiple compression engines or encoders was known in the art (Ex. 1003 ¶¶ 190–192), we are not satisfied that a person of skill in the art would have used a plurality of encoders to compress OS files given Settsu's disclosure. It appears that Petitioner proffers its arguments based on hindsight reasoning using the claim language as a roadmap for what a person of ordinary skill in the art could have done, rather than what they would have done. *See* Pet. 54–55.

Accordingly, having reviewed the entirety of the record and evidence, we determine Petitioner has established by a preponderance of the evidence that claims 35–45 and 97 of the '862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu. Additionally, we determine Petitioner has failed to establish by a preponderance of the evidence that claim 46 of the '862 patent is unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu.

F. Alleged Obviousness of the Challenged Claims in View of Settsu and Zwiegincew

Petitioner contends the combination of Settsu and Zwiegincew teaches or suggests each element of claims 5, 35–46, 97, 98, and 112. Pet. 16–60. Patent Owner disputes Petitioner’s contentions. PO Resp. 31–46. For reasons that follow, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that the challenged claims of the ’862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Zwiegincew.

1. *Analysis of Cited Art as Applied to Independent Claim 5*

Petitioner contends the teachings of Settsu and Zwiegincew renders every element of claim 5 of the ’862 patent obvious. Pet. 17–38. Patent Owner disputes Petitioner’s contentions. PO Resp. 31–36. Patent Owner specifically argues that Petitioner fails to show that the combined disclosures of Settsu and Zwiegincew teach “updating the boot data list.” *Id.* During oral argument, Patent Owner also argued that Petitioner failed to show a reason to combine Settsu and Zwiegincew. See Tr. 50. We address the issues raised by Patent Owner in more detail below.

a. *“updating the boot data list”*

As described above in § II.E.1., Petitioner contends Settsu describes a boot data list in at least two ways: (1) in the form of lists of boot data that are stored within the OS functional module files themselves; and (2) in the form of lists of boot data that are referenced by mini OS module 7 in the course of booting the OS. Pet. 19–21, 34 (citing Ex. 1003 ¶¶ 70, 77, 116; Ex. 1006, 5:39–51, 7:65–8:23, 13:49–65, 14:44–52, 16:7–17:62, Figs. 12, 20 (ST213)). Petitioner argues that Settsu’s description renders obvious updating both types of lists. *Id.* at 34 (citing Ex. 1003 ¶ 117). According to

Petitioner, Settsu's OS main body module is divided into "a plurality of functional modules" that "are separately stored as compressed files" in "file system 5 of boot device 3." *Id.* (citing Ex. 1006, 13:55–65). From this description, Petitioner argues that a person of ordinary skill in the art would have understood that OS modules, such as Settsu's, require updates when new, updated code becomes available. *Id.* (citing Ex. 1003 ¶ 117 (citing Ex. 1026, 110–112) (describing regular updates to the Windows 98 operating system)).

Petitioner then cites to Zwiegincew, arguing that Zwiegincew describes a boot data list in the form of a scenario file that is used in the context of preventing hard page faults during a system's boot process. *Id.* at 35 (citing Ex. 1003 ¶ 119; Ex. 1010, [57], 1:45–51, 1:5–3:55, 8:66–9:13, Fig. 1, 2). According to Petitioner, Zwiegincew also describes automatic processes for generating a scenario file to ensure that it includes "ordered copies of the pages that will likely be retrieved from disk due to one or more hard page faults during the [corresponding] hard page fault scenario," and the "[a]utomatically generated scenario files may be subject to subsequent refinement, i.e., they may be input into the pattern-matching algorithm." *Id.* at 36 (citing Ex. 1010, 6:29–67, 7:38–40). Based on this description, Petitioner argues that a person of ordinary skill in the art would have been motivated to update automatically Settsu's boot data lists based on patterns of requests received during Settsu's boot process. *Id.* (citing Ex. 1003 ¶¶ 120–121; Ex. 1010, 6:29–61, 7:24–40). A person of ordinary skill in the art would have been motivated, for example, to ensure that frequently requested boot data is loaded into memory 2, with corresponding updates to a scenario file. *Id.* (citing Ex. 1003 ¶ 121).

Petitioner supports its position with the Declaration of Dr. Neuhauser. *Id.* at 35–36 (citing Ex. 1003 ¶¶ 117, 121). Dr. Neuhauser testifies that the combination of Settsu and Zwiegincew render obvious updating the boot data list. Ex. 1003 ¶ 122. Dr. Neuhauser specifically opines that, to benefit from updates in Settsu’s system or simply to account for changes in user preference, a person of ordinary skill in the art would have understood that Settsu’s system updates the OS data stored in boot device 3. *Id.* ¶ 117. Dr. Neuhauser further testifies that to accomplish these updates, a person of ordinary skill in the art would have understood that updates occur to boot data lists in Settsu; namely, updates to lists of boot data included within OS functional module files stored in boot device 3, as well as updates to lists of boot data, such as function definition file 71 or OS main body module 8. *Id.* ¶ 117 (citing Ex. 1006, 5:39–51, 13:55–65, 14:44–52, 16:7–17:62, Figs. 12, 20 (ST213)). Dr. Neuhauser then testifies that one of ordinary skill would have been motivated, for example, to utilize pattern recognition in Settsu’s processes for loading data into memory 2, to ensure that frequently requested boot data is loaded into memory 2, with corresponding updates to a scenario file, such as in Zwiegincew. *Id.* ¶ 121.

Patent Owner contests Petitioner’s position, arguing Petitioner relies on impermissible hindsight analysis as the basis for arguing that Settsu discloses “updating the boot data list” as recited in claim 5. PO. Resp. 25–30. According to Patent Owner, Zwiegincew does not remedy the deficiencies of Settsu and it would not motivate a person of ordinary skill in the art to update boot data lists in Settsu’s systems. *Id.* at 31.

Patent Owner argues that a person of ordinary skill in the art would have understood that Zwiegincew’s teachings relate to managing virtual memory through swapping data pages into RAM from disk memory and

swapping data pages out of RAM into disk memory. *Id.* at 32 (citing Ex. 2008 ¶ 82). Patent Owner assert that “[b]ecause the use of these techniques require the presence of a fully initialized operating system kernel, they can be applied only after the operating system kernel has booted up, and thus, cannot relate to the accelerated loading of the operating system kernel itself or to otherwise managing the boot-up process.” PO Resp. 32 (citing Ex. 2008 ¶ 112). As such, Patent Owner argues that “[a person of ordinary skill in the art] would have understood that Zwiegincew’s teachings cannot be used to speed up the loading of an operating system because Zwiegincew’s techniques *require* the presence of a functioning operating system to work.” *Id.*

We do not agree with Patent Owner, because (1) we find that Settsu’s teachings render “updating a boot data list” obvious (*see supra*, Section II.E.), and (2) Zwiegincew specifically recites “boot” as an example of a hard page fault scenario (Ex. 1010, Abstract, 2:12–15). Additionally, we credit the testimony of Dr. Neuhauser that a person of ordinary skill in the art would have applied Zwiegincew’s pre-fetching techniques during the boot process in order to reduce the occurrence of hard page faults during that process, thereby improving boot speed. *See* Ex. 1003 ¶¶ 88–90. Thus, we find that the subject matter of the claim as a whole would have been obvious to a person of ordinary skill in the art in view of Settsu and Zwiegincew.

b. Alleged Rationale to Combine Asserted Art

Patent Owner argues that Petitioner fails to prove why a person of ordinary skill in the art would have been motivated to incorporate Zwiegincew’s scenario file technique into Settsu’s function definitional file 71 of lists internal to each function module. *See* Tr. 50. Petitioner, however, argues that a person of ordinary skill would have been motivated to combine

Settsu's system and compression/decompression scheme with Zwiegincew's compression and decompression teachings because Zwiegincew discloses techniques that a skilled artisan would have recognized as complementary to Settsu's and that would have reduced the time required for a system to boot. Pet. 28 (citing Ex. 1003 ¶ 87).

Dr. Neuhauser supports Petitioner's position. Specifically, Dr. Neahauser testifies that a skilled artisan would have recognized that the occurrence of hard page faults during Settsu's boot process would be reduced by adapting Settsu's system to pre-fetch, from boot device 3 into memory 2, pages that are expected to be requested during the boot process, in the manner described by Zwiegincew, and that this would further speed Settsu's boot process. Ex. 1003 ¶¶ 88–89.

Given Zwiegincew explicit disclosure of “boot” as an example of a hard page fault scenario (Ex. 1010, Abstract, 2:12–15), we are satisfied Petitioner has articulated sufficient reasoning with rational underpinning to combine the references as asserted. Accordingly, for reasons stated above, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that challenged claim 5 of the '862 patent is unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Zwiegincew.

2. Dependent Claims 35–45 and 97

The Petition sets forth detailed contentions and supporting evidence alleging that claims 35–45 and 97 are obvious in light of the teachings of Settsu and Zwiegincew. Pet. 38–60. For instance, for the “plurality of files” limitation of claims 35, 40, and 42, Petitioner relies on Settsu's teachings of an OS “main body module” that is “divided into a plurality of functional

modules” that “are separately stored as compressed files” in “file system 5 of boot device 3.” Pet. 38 (citing Ex. 1006, 2:5–67, 13:55–65). According to Petitioner, as supported by Dr. Neuhauser’s testimony, a person of ordinary skill in the art would have understood that a program, like the operating system loaded by Settsu into memory 2, typically includes a plurality of files. *Id.* (citing Ex. 1003 ¶ 131). With respect to the claims reciting compression or decompression, such as claims 37, 38, 39, 43, and 97, Petitioner identifies disclosures in Settsu that describe loading OS files from a boot device’s file system “into [a] memory 2 as compressed data,” and then decompressing the loaded OS files to speed up a computer system’s boot process, while arguing Zwiegincew also recognizes the benefit of using compression/decompression on boot data. *Id.* at 41–43 (citing Ex. 1006, at [57], 1:1–4, 1:51–57, 3:6–9, 13:49–15:5; Ex. 1003 ¶ 142; Ex. 1010, 8:66–9:13, Figs. 1, 2).

As for those claims that recite a form of dictionary encoding or the Lempel-Ziv algorithm for compression, such as claims 44 and 45, Petitioner notes that the Lempel-Ziv algorithm, which uses a form of dictionary encoding to compress data, was well-known at the time of the ’862 patent. *Id.* at 50 (citing Ex. 1003 ¶ 176). According to Petitioner, a skilled artisan would have found it obvious to use the well-known Lempel-Ziv algorithm to compress the OS files in the combination of Settsu and Zwiegincew. *Id.* at 50–51 (citing Ex. 1003 ¶¶ 175–177).

Patent Owner generally presents the same or similar arguments for claims 35–45 and 97 as presented for claim 5, which are unpersuasive on the present record for similar reasons as for claim 5. Accordingly, for reasons stated above, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that

challenged claims 35–45 and 97 of the '862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Zwiegincew.

a) Dependent Claim 46

Claim 46 recited “wherein a plurality of encoders was utilized to encode the compressed boot data.” Ex. 1001, 30:16–17.

Patent Owner contends that Settsu fails to teach the use of a “plurality of encoders” to “encode the compressed boot data.” PO Resp. 43–44. Specifically, Patent Owner argues the assertion that Settsu alone renders the use of a plurality of encoders is pure attorney argument unsupported by any evidence, including expert testimony. *Id.* According to Patent Owner, Dr. Neuhauser does not offer an opinion or a reasoned analysis that Settsu alone renders claim 46 obvious. *Id.* (citing Ex. 2011, 319:18–20 (Q: You do not provide an opinion that Claim 46 is obvious in view of Settsu alone, correct? A: That’s correct.)).

Petitioner contests Patent Owner’s position. Petitioner first acknowledges that, although “Settsu’s system stores operating system files in compressed form on boot device 3,” “Settsu fails to explicitly disclose a step of compressing the OS files.” Pet. 40 (citing Ex. 1003 ¶ 135). Yet, according to Petitioner, a person of ordinary skill in the art “would have understood that a step of compressing the OS files would have been needed to arrive at the OS files in compressed form.” *Id.* Petitioner argues that “Settsu simply omits mention of this necessary step and presupposes existence of the compressed OS files.” *Id.* Thus, Petitioner asserts that “a [person of ordinary skill in the art] would have understood and found obvious that a step of compressing the OS files occurs for Settsu’s OS files to be used in compressed form.” *Id.*

Petitioner further contends that “[t]o perform this compression, a [person of ordinary skill in the art] would have understood that a form of compression or encoding would have been used by a compressor or encoder to compress or encode the data.” Pet. 54. Petitioner relies on the testimony of Dr. Neuhauser to support its position. Dr. Neuhauser testifies that it was well-known at the time of the ’862 patent and Settsu to utilize a plurality of encoders to encode compressed data. Ex. 1003 ¶¶ 190–192 (citing Ex. 1008, 14:9–30, 8:43–19:30, 22:66–23:24, Figs. 10A, 10B, 13; Ex. 1009, 12:61–13:7, 13:52–56, 14:49–50, 29:13–14). Petitioner, thus, asserts that “from Settsu’s description of compressed OS files and the well-known solution of utilizing a plurality of encoders to encode compressed data, a [person of ordinary skill in the art] would have found it obvious to use a plurality of encoders to compress Settsu’s OS files and to provide the compressed OS files to boot device 3.” Pet. 54 (citing Ex. 1003 ¶¶ 190–192).

We agree with Petitioner that given the explicit disclosure in Settsu of storing compressed files and decompressing files, a person of ordinary skill in the art would have understood that at least compression and decompression would occur. We do not agree, however, that a person of ordinary skill in the art would have found it obvious from Settsu’s teachings, that multiple or a plurality of encoders would have been used to achieve the compression of files. Petitioner has not identified nor does not appear to be any suggestion or teaching in Settsu that would lead a person of ordinary skill in the art to use a plurality of encoders. Although we credit the testimony of Dr. Neuhauser that using multiple compression engines or encoders was known in the art (Ex. 1003 ¶¶ 190–192), we are not satisfied that a person of skill in the art would have used a plurality of encoder to compress OS files given Settsu’s disclosure. It appears that Petitioner

proffers its arguments based on hindsight reasoning using the claim language as a roadmap for what a person of ordinary skill in the art could have done, rather than what they would have done. *See* Pet. 54–55.

b) Dependent Claim 98

Claim 98 recites “disassociating non-accessed boot data from the boot data list.” Ex. 1001, 32:49–52.

Patent Owner argues that Petitioner incorrectly relies on Zwiegincew as teaching the limitation of claim 98. PO Resp. 37 (citing Pet. 58–59). Specifically, Patent Owner argues that Dr. Neuhauser’s reliance on and citation to column 6 lines 20–25 of Zwiegincew for this element is incorrect. *Id.* (citing Ex. 1003 ¶ 204). According to Patent Owner, Dr. Neuhauser testified that the cited passage does not address Zwiegincew’s scenario file and does not discuss boot data, but is a general discussion of memory management as performed by a virtual memory manager. *Id.* (citing Ex. 2011, 150:8–17).

Petitioner contests Patent Owner’s position, arguing that the Petition relies on more than one citation to Zwiegincew. Specifically, Petitioner argues that Zwiegincew’s automatic generation and refinement of scenario files based on pattern recognition ensures that pages corresponding to faults that re-occur frequently are included. Reply 17 (citing Pet. 58–59). Petitioner explains, and Dr. Neuhauser testifies that, Zwiegincew’s pattern recognition automatically generates scenario files with pages most likely to be accessed and then refines scenario files to include pages corresponding to faults that re-occur frequently while removing pages corresponding to faults that do not re-occur frequently. *Id.*; Ex. 1003 ¶¶ 203–205. According to Petitioner, “[w]ith this technique, Zwiegincew generates/refines scenario files by associating frequently-accessed pages (i.e., pages corresponding to

faults that re-occur frequently) and disassociating infrequently-accessed or non-accessed pages (i.e., pages corresponding to faults that do not re-occur frequently).” *Id.*

Crediting Dr. Neuhauser’s testimony regarding Zwiegincew assessing how scenario files are refined based on patent recognition, we agree with Petitioner and find that a person of ordinary skill in the art would have understood that, in refining scenario files, “requested pages are swapped with less used pages in the RAM 220” such that “the least recently used pages are” removed from the scenario file in RAM.” *See* Ex. 1003 ¶¶ 203–205; Reply 18.

c) Dependent Claim 112

Claim 112 recites “updating the boot data list in response to the utilizing.” Ex. 1001, 34:7–9.

Patent Owner argues that Petitioner incorrectly relies on Zwiegincew as teaching the limitation of claim 98. PO Resp. 41 (citing Pet. 58–59). Specifically, Patent Owner argues that Dr. Neuhauser’s reliance on and citation to column 6 lines 20–25 of Zwiegincew for this element is incorrect. *Id.* (citing Ex. 1003 ¶ 204). According to Patent Owner, Dr. Neuhauser testified that the cited passage does not address Zwiegincew’s scenario file and does not discuss boot data, but is a general discussion of memory management as performed by a virtual memory manager. *Id.* (citing Ex. 1003 ¶¶ 206–209, Pet. 58–60). Patent Owner further argues that Petitioner fails to explain (1) why general memory management functionality would motivate updating a Zwiegincew scenario file in response to utilizing the information in the scenario file or (2) how Zwiegincew’s automated refinement of scenario files would motivate a person of ordinary skill in the art to update a boot data list in response to

utilizing the information therein. *Id.* at 42–43 (citing Ex. 1003 ¶¶ 206–209, Pet. 58–59).

Petitioner contests Patent Owner’s position, arguing that a person of ordinary skill in the art would have been motivated by Zwiegincew to modify Settsu to update a scenario file, or boot data list, containing page data in response to utilizing certain pages from disk storage. Pet. 60. According to Petitioner, “[a person of ordinary skill in the art] would have found it obvious to update a scenario file, or boot data list, containing page data in response to utilizing certain pages accessed from disk storage during execution of processes, including booting a computer system.” *Id.* (citing Ex. 1003 ¶¶ 206-208; Ex. 1010, 2:10-15). Petitioner explains, and Dr. Neuhauser testifies that, Zwiegincew’s pattern recognition automatically generates scenario files with pages most likely to be accessed and then refines scenario files to include pages corresponding to faults that re-occur frequently while removing pages corresponding to faults that do not re-occur frequently. Reply 21 (citing Ex. 1003 ¶¶ 206–208). According to Petitioner, “[w]ith this technique, Zwiegincew generates/refines scenario files by associating frequently-accessed pages (i.e., pages utilized during boot) and disassociating infrequently-accessed or non-accessed pages (i.e., pages corresponding to faults that do not re-occur frequently) with scenario files to address potential page faults.” *Id.*

Crediting Dr. Neuhauser’s testimony regarding Zwiegincew assessing how scenario files are refined based on patent recognition, we agree with Petitioner and find that a person of ordinary skill in the art would have understood that, in refining scenario files, “requested pages are swapped with less used pages in the RAM 220” such that “the least recently used pages are” removed from the scenario file in RAM,” the boot data list would

be “refined” or “updated” in response to the data list being used. *See* Ex. 1003 ¶¶ 203–205; Reply 18.

d) Conclusion

Having reviewed the entirety of the record and evidence, we determine Petitioner has established by a preponderance of the evidence that claims 5, 35–45, 97, 98, and 112 of the ’862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Zwiegincew. Additionally, we determine Petitioner has failed to establish by a preponderance of the evidence that claim 46 of the ’862 patent is unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Zwiegincew.

G. Alleged Obviousness of the Challenged Claims in in View of Settsu and Dye

Petitioner contends the combination of Settsu and Dye teaches or suggests each element of claims 5, 35–46, and 97. Pet. 60–70. Patent Owner disputes Petitioner’s contentions. PO Resp. 43–51. Patent Owner specifically argues that Dye only discloses a single encoder. *Id.* During oral argument, Patent Owner also argued that Petitioner failed to show a reason to combine Settsu and Dye. *See* Tr. 50. We address the issues raised by Patent Owner in more detail below.

Having reviewed the entirety of the record and evidence, we determine Petitioner has established by a preponderance of the evidence that claims 5, 35–46, 97, 98, and 112 of the ’862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Dye.

1. Analysis of Cited Art as Applied to Claims 5, 35–45, 97, 98, and 112

Petitioner contends the combined teachings of Settsu and Dye render every element of claims 5, 35–45, 97, 98, and 112 of the ’862 patent

obvious. Pet. 60–68. With regards to independent claim 5, Petitioner contends that the combined teachings of Settsu and Dye meet this limitation because Settsu describes “a method of booting an information processing apparatus at a high speed.” Pet. 37 (citing Ex. 1006, [57], 1:8–12, 1:43–2:25, 3:6–25, 10:43–12:16, 13:49–15:4; Ex. 1003 ¶ 27). Specifically, Petitioner argues that Settsu’s mini OS module 7’s OS loading and decompression processing module 50 “decompresses each of the plurality of functional modules each time it loads each of them into memory” and that, “[a]s a result, the time required for I/O processing can be reduced.” *Id.* (citing Ex. 1006, 14:64–67)(emphasis omitted). According to Petitioner, Settsu’s use of compression/decompression “provides an advantage of being able to further reduce the time required for booting up the information processing apparatus.” *Id.* (citing Ex. 1006, 15:1–4; Ex. 1003 ¶ 124). Because Settsu’s use of compression/decompression further reduces the time required for booting up, Petitioner argues that Settsu’s loading, accessing, and decompressing necessarily occurs within a period of time which is less than a time to access the boot data in an uncompressed form. *Id.* at 37; *see* Ex. 1003 ¶¶ 123–126.

The Petition sets forth detailed contentions and supporting evidence alleging that claims 35–46 and 97 are obvious in light of the teachings of Settsu and Dye. Pet. 65–70. For instance, for the claims reciting compression or decompression, such as claims 37, 38, 39, 43, and 97, Petitioner identifies disclosures in Settsu that describe loading OS files from a boot device’s file system “into memory 2 as compressed data,” and then decompressing the loaded OS files to speed up a computer system’s boot process, while Dye provides further motivation to use Settsu’s OS functional modules on boot device 3 in compressed form. *Id.* at 65–66 (citing Ex. 1003

¶ 138). As for those claims that recite a form of dictionary encoding or the Lempel-Ziv algorithm for compression, such as claims 44 and 45, Petitioner notes that the Lempel-Ziv algorithm, which uses a form of dictionary encoding to compress data, was well-known at the time of the '862 patent. *Id.* at 67–68 (citing Ex. 1003 ¶ 180). According to Petitioner, a skilled artisan would have found it obvious to use the well-known Lempel-Ziv algorithm to compress the OS files in the combination of Settsu and Dye. *Id.* at 67–68 (citing Ex. 1003 ¶¶ 181, 188–189).

Patent Owner disputes Petitioner's contentions. PO Resp. 43–51. Patent Owner generally presents the same or similar arguments for the challenge based on Settsu and Dye as presented for the challenge based on Settsu alone, which are unpersuasive on the present record for the same reasons discussed in detail with regards to Settsu. *See supra*, Section II.E. Accordingly, for reasons stated above, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that challenged claims 5, 35–45, 97, 98, and 112 of the '862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Dye.

2. Analysis of Cited Art as Applied to Claim 46

Dependent claim 46 each requires “a plurality of encoders was utilized to encode the compressed boot data.” Ex. 1001, 30:16–17.

Petitioner contends the combined teachings of Settsu and Dye render every element dependent claim 46 of the '862 patent obvious. Pet. 67–70; Reply 21–26. Patent Owner disputes Petitioner's contentions. PO Resp. 45–51. Patent Owner specifically argues that Dye does not disclose a “plurality of encoders” but rather uses a single algorithm that is distributed among several stages. *Id.* at 46–47 (citing Ex. 2008 ¶ 130). According to

Patent Owner, each stage is not a separate encoder but instead, each unit is a part of Dye's single encoder. *Id.* at 47–48 (citing Ex. 1008, 4:20–28).

Petitioner contends Dye teaches a “plurality of encoders” as part of its compression/decompression engine. Pet. 68–69. Petitioner relies on the declaration of Dr. Neuhauser to support its position. Dr. Neuhauser testifies that Dye uses a “parallel lossless compression/decompression” “designed to process stream data at more than a single byte or symbol (character) at one time.” Ex. 1003 ¶ 195 (citing Ex. 1009, 4:9–30). Dr. Neuhauser explains that Dye's compression algorithm analyzes multiple symbols in parallel, and provides multiple compressed outputs in parallel, and provides multiple compressed outputs in parallel. *Id.* ¶ 195 (citing Ex. 1008, 18:43–19:30, 22:66–23:24, Fig. 13; Ex. 1009, 12:61–13:7, 13:52–56), ¶¶ 196–197.

Petitioner explains that Figure 13 of Dye “serves ‘as an encoder of the Input Stream’ ‘because this logic **encodes** four bytes of the input stream 610 against the four bytes in the history table” and Dye further discloses in Figure 13 logic, for one stream – “there are 16 such encoders used in the system of Dye. . . .” Reply 23 (citing Ex. 1003 ¶¶ 43–61, 194–198).

Having reviewed Dye, we find that it teaches, in Figure 10B, entries A, B, C, and D. *See* Ex. 1008, Fig. 10B. Specifically, Figure 13 teaches processing using entry D, and, as such, logic shown in Figure 13 would be present for processing each of entries A, B, and C. *Id.* at Figs. 10B, 13. Additionally, we interpret “encoder” as “any hardware or software that converts information to a particular form or format.” In light of this interpretation, we determine Dye teaches a plurality of encoders. Moreover, we credit the testimony of Dr. Neuhauser that Dye teaches the logic shown in Figure 13 would be present for each Data Entry (e.g., Data Entry A, B, C, D) and thus, teaches a plurality of encoders. *See* Ex. 1003 ¶¶ 43–61, 194–

198; Ex. 1008, Figs. 10B, 13. We further find Dye teaches these encoders may be in a parallel configuration. *See* Ex. 1008, 18:60–63, 22:64–23:4, Figs. 10B, 13. Thus, based on the entirety of the record before us, we find Dye teaches “a plurality of encoders was utilized to encode the compressed boot data.” as recited in claim 46. It follows, Petitioner has established by a preponderance of the evidence that claim 46 of the ’862 patent is unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Dye

3. Alleged Rationale to Combine Asserted Art

Patent Owner argues that Petitioner fails to articulate a sufficient motivation to combine Settsu and Dye without the benefit of impermissible hindsight analysis. *See* Tr. 50. Petitioner, however, argues that a person of ordinary skill would have been motivated to combine Settsu’s system with Dye’s compression and decompression teachings because Dye indicates those teachings lead to improved memory capacity and performance, specifically improved data density, efficiency, and bandwidth. Pet. 61–62 (citing Ex. 1003 ¶¶ 92–97; Ex. 1008, at [57], 2:42–4:55, 7:34–9:5, 10:19–41). Petitioner further argues that through this incorporation, Dye confirms that the controller “may couple to any of various types of memory, as desired,” enabling it to “greatly increas[e] the performance of the computer system.” *Id.* at 62 (citing Ex. 1009, 10:19–41).

Petitioner additionally argues that a person of ordinary skill in the art would have been motivated by Dye to modify Settsu to perform compression/decompression on data stored in Settsu’s memory devices at a rate that speeds up data access time from those devices. *Id.* at 64–65 (citing Ex. 1003 ¶ 127). Petitioner cites to Dye’s explanation that its “fast parallel compression and decompression technology . . . increase[s] the effective density and read access time of non-volatile storage devices,” including hard

disk drives and other memory devices, effectively “matching the data access speeds . . . to that of the microprocessor.” *Id.* at 65 (citing Ex. 1008, at [57], 7:34–43).

Petitioner relies on the testimony of Dr. Neuhauser to support its position. Dr. Neuhauser opines that a person of ordinary skill in the art would have understood that modifying Settsu to perform compression/decompression on data stored in Settsu’s memory devices at a rate that speeds up data access time from those devices would further speed Settsu’s boot process. Ex. 1003 ¶ 127. Petitioner, thus, concludes that Settsu and Dye render obvious that the loading, the accessing, and the decompressing occur within a period of time which is less than a time to access the boot data from the first memory if the boot data was stored in the first memory in an uncompressed form. Pet. 65.

Despite Patent Owner’s arguments, we agree with Petitioner’s position, as supported by the testimony of Dr. Neuhauser. Specifically, given Settsu’s explicit disclosure of loading/storing of compressed files and then decompressing loaded files, we find Petitioner has articulated sufficient reasoning with rational underpinning to combine Settsu with Dye’s parallel decompression/compression system.

4. Conclusion

Accordingly, for the foregoing reasons, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that the challenged claims 5, 35–46 and 97 of the ’862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu and Dye.

H. Alleged Obviousness of the Challenged Claims in View of Settsu, Zwiegincew, and Dye

Petitioner contends the combination of Settsu, Zwiegincew, and Dye teaches or suggests each element of claims 5, 35–46, 97, 98, and 112. Pet. 60–70. Patent Owner disputes Petitioner’s contentions. PO Resp. 43–51. Petitioner relies on these references for the same teachings as discussed above for the other grounds asserted in the Petition. *See* Pet. 60–70. In addition, Petitioner contends a person of ordinary skill would have had reason to combine these references based on similar arguments and evidence as for those other grounds. *Id.* at 60–64. Patent Owner also relies on arguments similar to those it advances for the other asserted grounds. *See* PO Resp. 43–51. For the same reasons discussed above, we determine based on the entirety of the record before us, Petitioner has established by a preponderance of the evidence that the challenged claims 5, 35–46, 97, 98, and 112 of the ’862 patent are unpatentable under 35 U.S.C. § 103(a) as obvious over Settsu, Zwiegincew, and Dye.

G. Remaining Patent Owner Arguments

Patent Owner makes additional arguments stating that post grant review proceedings, such as this trial, are unconstitutional and are an impermissible taking of a private right without Article III oversight. PO Resp. 51–53. We decline to consider the constitutional challenge as, generally, “administrative agencies do not have jurisdiction to decide the constitutionality of congressional enactments” where consideration of the constitutional question would “require the agency to question its own statutory authority or to disregard any instructions Congress has given it.”

Riggin v. Office of Senate Fair Employment Practices, 61 F.3d 1563, 1569–70 (Fed. Cir. 1995).⁵

II. MOTION TO EXCLUDE EVIDENCE

Patent Owner also filed Objections to Evidence in Petitioner’s Reply (Paper 18) and then a Motion (Paper 25) seeking to exclude evidence presented by Petitioner. Specifically, Patent Owner seeks to exclude Exhibits 1038–1040 and 1044. Paper 25, 1. Because our Decision does not rely on the challenged exhibits, we dismiss Patent Owner’s motion as moot.

III. CHALLENGE TO PETITIONER’S REPLY ARGUMENTS

Patent Owner filed a list of alleged improper reply arguments (Paper 19) to which Petitioner filed a Reply (Paper 20). Patent Owner lists several portions of Petitioner’s Reply and evidence allegedly beyond the scope of what can be considered appropriate for a reply. *See* Paper 19.

We have considered Patent Owner’s listing, but disagree that the cited portions of Petitioner’s Reply and reply evidence are beyond the scope of what is appropriate for a reply. Replies are a vehicle for responding to arguments raised in a corresponding patent owner response. Petitioner’s arguments and evidence that Patent Owner objects to are not beyond the proper scope of a reply because we find that they fairly respond to Patent Owner’s arguments raised in Patent Owner’s Response. *See Idemitsu Kosan Co., LTD. v. SFC Co. LTD*, 870 F.3d 1376, 1381 (“This back-and-forth

⁵ On June 12, 2017, the United States Supreme Court granted a petition for a writ of certiorari in *Oil States Energy Services, LLC v. Green’s Energy Group, LLC*, 639 Fed. App’x 639 (Fed. Cir 2016), cert. granted, 198 L. Ed. 2d 677 (U.S. Jun. 12, 2017) (No. 16-712). The Court will answer the question of whether the USPTO’s statutorily created IPR process is unconstitutional.

shows that what Idemitsu characterizes as an argument raised ‘too late’ is simply the by-product of one party necessarily getting the last word. If anything, Idemitsu is the party that first raised this issue, by arguing—at least implicitly—that Arakane teaches away from non-energy-gap combinations. SFC simply countered, as it was entitled to do.”). We also note for each of the items in Patent Owner’s list, Petitioner cites to pages in the Petition for support. *See e.g.*, Paper 20, 2 (“Support for Petitioner’s argument can be found at, for example, pages 58–59 of the Petition.”), 4 (“Support for Petitioner’s argument can be found at, for example, pages 19–30 of the Petition.”), 5 (“Support for Petitioner’s argument can be found at, for example, pages 34–37 of the Petition.”), 6 (“Support for Petitioner’s argument can be found at, for example, pages 54–55 of the Petition.”), 7 (“Support for Petitioner’s argument can be found at, for example, pages 68–70 of the Petition.”).

IV. CONCLUSION

For the foregoing reasons, we conclude Petitioner has shown by a preponderance of the evidence that (1) Claims 5, 35–45, and 97 are unpatentable under 35 U.S.C. § 103(a) in view of Settsu; (2) Claims 5, 35–45, 97, 98, and 112 as unpatentable under 35 U.S.C. § 103(a) in view of Settsu and Zwiegincew; (3) Claims 5, 35–46, and 97 as unpatentable under 35 U.S.C. § 103(a) in view of Settsu and Dye; and (4) Claims 5, 35–46, 97, 98, and 112 as unpatentable under 35 U.S.C. § 103(a) in view of Settsu, Zwiegincew, and Dye.

V. ORDER

Accordingly, it is

ORDERED that, by a preponderance of the evidence, claims 5, 35–46, 97, 98, and 112 of the '862 patent are unpatentable;

FURTHER ORDERED that Patent Owner's Motion to Exclude is *dismissed as moot*; and

FURTHER 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.

IPR2016-01739
Patent 8,880,862 B2

PETITIONER:

Walter Karl Renner
Jeremy J. Monaldo
Katherine Lutton
Andrew Patrick
FISH & RICHARDSON P.C.
axf-ptab@fr.com
jjm@fr.com
lutton@fr.com
patrick@fr.com

PATENT OWNER:

Joseph F. Edell
Richard Z. Zhang
FISCH SIGLER LLP
joe.edell.ipr@fischllp.com
richard.zhang.ipr@fischllp.com