

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

SECURITY FIRST INNOVATIONS, LLC,
Patent Owner.

Case IPR2024-00213
Patent 9,338,140

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. OVERVIEW OF THE '140 PATENT AND THE CHALLENGED CLAIMS.	2
III. OPHIR-TAKEDA DOES NOT RENDER THE CLAIMS OBVIOUS (ALL CLAIMS, GROUNDS 1-3).....	3
A. Ophir And Takeda Do Not Disclose The Claimed Inventions.	3
B. Ophir-Takeda Does Not Disclose or Render Obvious A “Plurality Of Shares” “Stored On” “A Plurality Of Physical Storage Devices” (Limitations 1[a], 1[b][2]).	4
C. A POSITA Would Not Be Motivated To Combine Ophir-Takeda. ...	15
1. Petitioner’s Proposed Modification Of Ophir Based On Takeda Directly Conflicts With, And Undermines The Goal And Method Of Operation Of, Ophir.	15
2. The Petition Fails To Show That A POSITA Would Be Motivated To Modify Ophir To Store Blocks From A Single Dataset Amongst Multiple Storage Devices.....	21
D. Ophir Does Not Disclose And/Or Render Obvious “A Directory Mapped To The Plurality Of Physical Storage Devices.”	24
1. Ophir’s Mediator Does Not Disclose Or Render Obvious The Required Directory.	24
2. Ophir Does Not Disclose The Required Directory By Disclosing “Block Services.”	27
3. Ophir Does Not Disclose The Required Directory By Disclosing “File Services.”	31

IV. DICKINSON-TAKEDA DOES NOT DISCLOSE OR RENDER OBVIOUS THE CLAIMS OF THE '140 PATENT (ALL CLAIMS, GROUND 4).	33
A. The Claims Are Directed To Different Technology Than Dickinson.....	33
1. The Trust Engine Of Dickinson Is A Fundamentally Different Invention Than The Virtual Disk Of Claim 1.	33
2. The '140 Patent Adds Critical New Material Regarding The Limitations Of Claim 1 That Is Not Disclosed In Dickinson. ..	39
a. New Disclosures In The '140 Patent Not Disclosed In Dickinson Teach Limitation 1[b][1].	39
b. New Disclosures In The '140 Patent Not Disclosed In Dickinson Teach Limitation 1[B][3].	41
B. Dickinson Was Expressly Considered By The Examiner During Prosecution And Should Not Be Revisited.	44
C. Dickinson-Takeda Does Not Disclose Or Render Obvious Limitation 1[B][1].	47
1. Dickinson Does Not Disclose A “Directory.”	48
2. Dickinson Does Not Disclose A Directory “Mapped To The Plurality Of Physical Storage Devices.”	49
3. The “Physical Locations Of The Shares Are Hidden From The Client Device.”	53
4. Dickinson Does Not Disclose “A Secure Storage System Configured To[] Present To A Client Device A Virtual Disk.”	55
D. Dickinson-Takeda Does Not Disclose Or Render Obvious Limitation 1[b][3].	61

1.	Dickinson Alone Does Not Disclose Or Render Obvious Limitation 1[b][3].	61
2.	The Combination Of Dickinson And Takeda Does Not Disclose Or Render Obvious Limitation 1[b][3].	63
V.	CONCLUSION	67

TABLE OF AUTHORITIES

Page(s)

COURT DECISIONS

<i>3M Co. v. Evergreen Adhesives, Inc.</i> , 860 Fed. App'x. 724 (Fed. Cir. 2021)	26
<i>Arctic Cat Inc. v. Polaris Indus.</i> , 795 Fed. App'x. 827 (Fed. Cir. 2019)	21, 64
<i>Arendi S.A.R.L. v. Apple, Inc.</i> , 832 F.3d 1355 (Fed. Cir. 2016)	10
<i>Belden Inc. v. Berk-Tek LLC</i> , 805 F.3d 1064 (Fed. Cir. 2015)	27
<i>DSS Tech. Mgmt. v. Apple Inc.</i> , 885 F.3d 1367 (Fed. Cir. 2018)	10
<i>Henny Penny Corp. v. Frymaster LLC</i> , 938 F.3d 1324 (Fed. Cir. 2019)	21, 64
<i>Institut Pasteur v. Focarino</i> , 738 F.3d 1337 (Fed. Cir. 2013)	21
<i>K/S HIMPP v. Hear-Wear Techs., LLC</i> , 751 F.3d 1362 (Fed. Cir. 2014)	10
<i>In re Magnum Oil</i> , 829 F.3d 1364 (Fed. Cir. 2016)	13, 49
<i>Medichem, S.A. v. Rolabo, S.L.</i> , 437 F.3d 1157 (Fed. Cir. 2006)	22
<i>Metalcraft of Mayville, Inc. v. Toro Co.</i> , 848 F.3d 1358 (Fed. Cir. 2017)	14, 15

<i>Microsoft Corp. v. Biscotti, Inc.</i> , 878 F.3d 1052 (Fed. Cir. 2017)	32
<i>In re NTP, Inc.</i> , 654 F.3d 1279 (Fed. Cir. 2011)	5
<i>Plas-Pak Indus., Inc. v. Sulzer Mixpac AG</i> , 600 F. App'x. 755 (Fed. Cir. 2015)	20
<i>Polaris Indus., Inc. v. Arctic Cat, Inc.</i> , 882 F.3d 1056 (Fed. Cir. 2018)	21
<i>Princeton Biochemicals, Inc. v. Beckman Coulter, Inc.</i> , 411 F.3d 1332 (Fed. Cir. 2005)	15
<i>In re Ratti</i> , 270 F.2d 810 (CCPA 1959)	20
<i>Ruiz v. A.B. Chance Co.</i> , 357 F.3d 1270 (Fed. Cir. 2004)	15
<i>SIMO Holdings, Inc. v. H.K. uCloudlink Network Tech., Ltd.</i> , 983 F.3d 1367 (Fed. Cir. 2021)	5
<i>Trintec Indus., Inc. v. Top-U.S.A. Corp.</i> , 295 F.3d 1292 (Fed. Cir. 2002)	25, 32
<i>Virtek Vision Int'l. ULC, v. Assembly Guidance Systems, Inc.</i> , ___ F.4th ___, Case No. 22-1998 slip op. (Fed. Cir. Mar. 27, 2024) .	14, 25, 30, 32

AGENCY DECISIONS

<i>Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte GmbH</i> , IPR2019-01469, Paper 6 (Feb. 13, 2020) (precedential).....	44, 47
<i>Edwards Lifesciences Corp. v. Aortic Innovations LLC</i> , IPR2023-01232, Paper 10 (Feb. 7, 2024).....	26, 29
<i>Parsons Xtreme Golf, LLC v. Taylor Made Golf Co., Inc.</i> , IPR2018-00518, Paper 14 (July 18, 2018)	26, 29

RPX Corp. v. Parity Networks, LLC,
IPR2018-00097, Paper 7 (Apr. 24, 2018)..... 26, 29

Xerox Corp. v. Bytemark, Inc.,
IPR2022-00624, Paper 9 (Aug. 24, 2022) (precedential)..... 13, 49, 55

STATUTES

35 U.S.C. § 325(d) 44

REGULATIONS

37 C.F.R. § 42.6(a)(3) 26

37 C.F.R. § 42.65 13, 49, 55

EXHIBIT LIST	
2001	Reserved
2002	Reserved
2003	Declaration of Aviel D. Rubin, Ph.D.
2004	Reserved
2005	Bruce Schneier Curriculum Vitae [Schneier-CV]
2006	Bruce Schneier, <i>Why Cryptography Is Harder Than It Looks</i> , SCHNEIER ON SECURITY, 1997, https://www.schneier.com/essays/archives/1997/01/why_cryptography_is.html [Schneier-Why-Cryptography-Is-Harder-Than-It-Looks]

I. INTRODUCTION

The Petition should be denied as it fails to demonstrate a reasonable likelihood of prevailing on at least one challenged claim.

Grounds 1-3 are centered upon Ophir and Takeda, both of which are far afield of the claims. Whereas the claimed invention focuses upon dividing a dataset into shares and storing those shares amongst multiple physical storage devices, neither Ophir nor Takeda, nor the combination of the two, suggest any such thing. *See* Section III.B. Petitioner, moreover, fails to demonstrate that a POSITA would be motivated to modify Ophir in view of Takeda to this end, particularly when doing so would, *inter alia*, eviscerate Ophir's goal and principle of operation. *See* Section III.C.

Petitioner's fourth ground, based upon the combination of Dickinson and Takeda, is even weaker. Dickinson was a major focus in prosecution and the Examiner correctly found that Dickinson did not satisfy two different claim limitations. *See* Section IV.B. And, even if Dickinson were to be revisited, Petitioner's recycled arguments should again be rejected. *See* Section IV.C-D.1. Petitioner's combination with Takeda, moreover, cannot salvage Dickinson, and Petitioner fails to prove a POSITA would even be motivated to make the combination. *See* Section IV.D.2.

II. OVERVIEW OF THE '140 PATENT AND THE CHALLENGED CLAIMS.

The '140 is directed towards an “ever-increasing need” to ensure stored data cannot be read or otherwise compromised by unauthorized parties. Ex. 1001 [’140] 1:22-29. To address this need, the '140 “provide[s] a method for securing virtually any type of data from unauthorized access or use” which includes, first, “parsing, splitting and/or separating the data to be secured into two or more parts or portions,” identified as shares. *Id.*, 2:33-37. The '140 then teaches the significant security advantages derived from storing those shares amongst multiple physical storage devices. Ex. 1001 [’140] 95:15-18, 21:4-12, 69:16-22, 53:6-18, 58:64-59:6, 6:43-51, 96:7-9, 59:27-34, 83:12-30; *see also* Ex. 2003 [Rubin-Decl.] ¶44. In addition, the dataset is further secured using a “session key,” and “data indicative of the at least one session key” is included with each of the shares. *Id.*, 98:2-22; 55:56-56:41.

The claims additionally require a directory, mapped to a plurality of physical storage devices, to manage the shares associated with a given dataset. *Id.*, 98:2-22; 87:16-34; Section IV.A.2(a). The directory provides client systems with a view of stored datasets but hides the underlying distribution of the shares among the storage devices from the client device. *Id.*, 87:35-59. In this way, an interface is provided that allows client systems to access the entirety of a particular dataset

without knowledge of the location of its individual shares. This enhances security by reducing the likelihood that an attacker with access to a client system can gain access to enough shares to reconstitute a dataset. *See* Ex. 2003 [Rubin-Decl.] ¶¶ 36-38.

III. OPHIR-TAKEDA DOES NOT RENDER THE CLAIMS OBVIOUS (ALL CLAIMS, GROUNDS 1-3).

A. Ophir And Takeda Do Not Disclose The Claimed Inventions.

In contrast to the claimed inventions which, *inter alia*, require storing shares of a dataset in more than one physical storage device (*see* Section II), Ophir and Takeda concern very different technologies. As Dr. Aviel Rubin¹ explains:

Ophir discloses a “mediator” that provides an interface between networked storage devices and client devices using a variety of protocols. “The mediator functions as a central point” for the encryption of datasets from the data clients to be stored on the storage devices, as well as decryption of the encrypted dataset retrieved from the storage devices for delivery to the data clients. Ex. 1007 [Ophir] Abstract. The mediator makes storage available for datasets provided by the clients, without requiring the clients to be involved in encryption or decryption operations, or burdened with key management. *Id.* Nothing in Ophir indicates that the mediator divides the datasets into shares, or stores a particular dataset on more than one storage device.

¹ Dr. Rubin is Professor Emeritus at John Hopkins University.

Takeda discloses a “ciphering apparatus” in which a plaintext dataset provided by a client is divided into blocks and the blocks are individually encrypted. Ex. 1008 [Takeda] Abstract. All of the separately-encrypted blocks are stored together in a single file on a single storage device, either with or without information about the keys or other techniques used to encrypt them. *See, e.g.*, Ex. 1008 [Takeda] ¶56.

Thus, in contrast to the ’140, neither Ophir nor Takeda concern dividing a dataset into shares *and* storing those shares amongst multiple storage devices.

Ex. 2003 [Rubin-Decl.] ¶¶39-41.

Petitioner asserts that “[a] POSA would have modified Ophir to implement Takeda’s ciphering/deciphering method in Ophir’s mediator.” Pet., 17. But, as explained below, Petitioner fails to demonstrate that the combination would disclose or render obvious numerous limitations or that a POSITA would be motivated to make the combination.

B. Ophir-Takeda Does Not Disclose or Render Obvious A “Plurality Of Shares” “Stored On” “A Plurality Of Physical Storage Devices” (Limitations 1[a], 1[b][2]).

Limitation 1[a] requires “a plurality of physical storage devices storing thereon a plurality of shares, the plurality of shares being associated with at least one session key used to secure a dataset.” Limitation 1[b][2] requires “generat[ing] the plurality of shares ... and distributing the dataset in the shares.”

These limitations together, therefore, require storing two or more shares generated from a dataset on two or more physical storage devices. The Ophir-Takeda combination neither discloses nor renders obvious this requirement.

It is well-established that “[t]he phrase ‘a plurality of’ means ‘at least two of.’” *SIMO Holdings, Inc. v. H.K. uCloudlink Network Tech., Ltd.*, 983 F.3d 1367, 1377 (Fed. Cir. 2021) (citations omitted); *In re NTP, Inc.*, 654 F.3d 1279, 1301-1302 (Fed. Cir. 2011) (assembling “‘information received from a plurality of the originating processors ... into a packet’” not met by reference that “does not disclose that a single packet may contain originated information from a plurality of originating processors.”). Because limitation 1[a] requires “[two or more] physical storage devices storing thereon [two or more] shares” and 1[b][2] requires “generat[ing] the [two or more] shares for storage on the [two or more] physical storage devices,” the claim, on its face, would not be satisfied by the plurality of shares being stored on a single physical storage device.

These limitations reflect a central feature of the ’140’s teachings, that dividing a dataset into shares and storing those shares amongst multiple storage devices helps secure the data. As Dr. Rubin explains:

Limitations 1[a] and 1[b][2] require a “plurality of shares,” *i.e.*, two or more shares be stored on a “plurality of physical storage devices,” *i.e.*, two or more storage devices.

These limitations reflect the '140's teachings that distributing shares amongst multiple storage devices substantially increases the security of sensitive information. Ex. 1001 ['140] 95:15-18 (“The sensitive information ... may be parsed and distributed to multiple locations...”); 21:4-12 (“the data splitting process 800 advantageously places portions of the sensitive data in each of the four data storage facilities ... such that no single data storage facility ... includes sufficient encrypted data to recreate the original sensitive data S”); 69:16-22; 53:6-18; 56:37-40; 58:64-59:6; 60:43-51; 96:6-9; 59:27-34; 83:12-30.²]

Ex. 2003 [Rubin-Decl.] ¶¶43-44. And, as Dr. Rubin explains, the Examiner (in prosecution), and the parties here have uniformly recognized that the claims require storing shares on different physical storage devices:

The prosecution history, consistent with the plain meaning of the claims, reinforces the requirement that the plurality of shares be stored on two or more storage devices. There, the Examiner found that limitation 1[a]'s requirement of “a plurality of physical storage devices” was allegedly satisfied by Dickinson's disclosure of data storage facilities D1, D2, D3, and D4 and the requirement of “storing thereon a plurality of shares” was allegedly taught by a disclosure that “the data splitting process 800 advantageously places portions of the sensitive data in each of the four data storage facilities D1 through D4, such that no single data storage facility D1 through D4 includes

² Certain parenthetical quotes omitted.

sufficient encrypted data to recreate the original sensitive data S.”
See Ex. 1002 [Pros.-Hist.] 1236-1237 (quoting Ex. 1005 [Dickinson] 21:10-14).

This understanding of the claim is also undisputed by the parties. Petitioner contends that Dickinson meets limitation 1[a] because the “encryption results (‘B’ and ‘D’) [are stored] **on at least two** of storage devices D1-D4.” Pet., 68. Similarly, in district court, Patent Owner’s infringement contentions rely on Petitioner’s storage system being designed such that “the dataset is broken up into shares for storage” and “**will not be stored entirely in a single server**, but instead will be broken up into chunks’ and **stored across different servers.**” Ex. 1044 [PO-Infringement-Contentions] 34.

Thus, the plain meaning of the claims, the specification, and the prosecution history are consistent in requiring that a dataset be divided into two or more shares which must be stored on two or more storage devices. There is further no dispute between the parties that the claims require that the two or more shares must be stored on two or more storage devices. The claim limitations are not satisfied if the plurality of shares are generated, but only stored on a single storage device.

Ex. 2003 [Rubin-Decl.] ¶¶46-48.

Critically, however, neither Ophir nor Takeda suggests storing shares of a given dataset on different physical storage devices. Petitioner contends that Takeda’s blocks comprise a “plurality of shares” and that these shares would be

stored on Ophir's "plurality of storage devices": "Ophir-Takeda stores the plurality of shares [Takeda's ciphered blocks] on [Ophir's] plurality of storage devices."

Pet., 31. But neither reference teaches or suggests two or more shares from a dataset being stored on two or more physical storage devices.

As Dr. Rubin explains, Takeda's blocks are stored together in a single storage device:

Even assuming that Takeda's blocks are "shares," Takeda does not teach or suggest storing its blocks/shares on two or more physical storage devices. Rather, Takeda's encrypted blocks (aka "ciphertext") are all stored in a single place (*e.g.*, "storage device 37" or "removable recording medium 39"). Ex. 1008 [Takeda] ¶¶91-92 ("The ciphertext is stored in the removable recording medium 39 or the storage device 37"), ¶¶7-8 (encrypted blocks are ciphertext).

Takeda does not teach storing two or more blocks of a dataset on two or more physical storage devices. Instead, Takeda suggests the opposite, indicating that its blocks are stored together. For example, Takeda's Figures 1B, 2B, 3A, and 3B, below, each show configurations of the divided dataset wherein the blocks are shown as stored together, both before ciphering (Fig. 1B) and after ciphering (Figs. 2B, 3A, 3B):

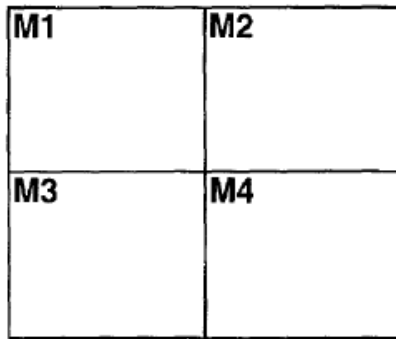


FIG.1B

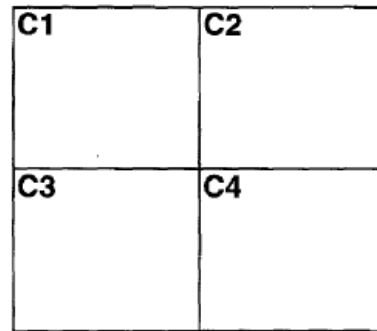


FIG.2B

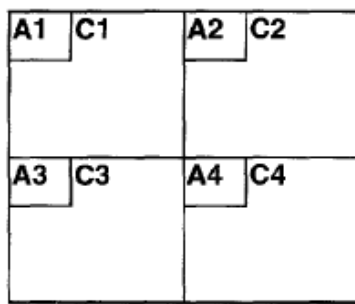


FIG.3A

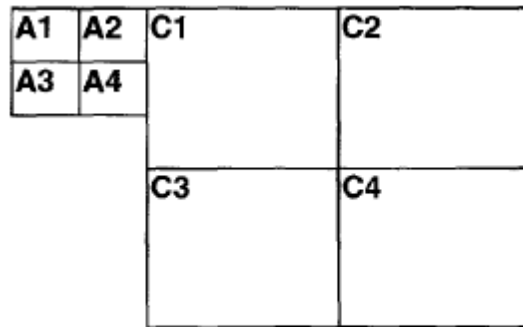


FIG.3B

This accords with Takeda’s written description which refers to the ciphered blocks collectively as a “ciphertext data group.” *Id.*, ¶48.

Ex. 2003 [Rubin-Decl.] ¶¶51-53.

Ophir cannot fill this gap. Ophir discloses multiple storage devices but does **not** disclose storing parts or portions of a single dataset on two or more physical storage devices. As Dr. Rubin explains:

Ophir concerns a secure storage network with a “secure storage data mediator” that receives and stores information from multiple clients across multiple storage devices. Ex. 1007 [Ophir] ¶¶11-13, Fig. 4. Ophir’s goal is to provide a storage network that “offers both the flexibility, expandability and scalability of [a] network, but which also places no encryption burdens on the data clients.” *Id.*, ¶8; *see also id.*, ¶¶9, 15.

But Ophir does **not** disclose or suggest, at all, dividing a dataset into pieces or anything else that might correspond to the “shares” of claim 1, much less storing those pieces on two or more physical storage devices.

Ex. 2003 [Rubin-Decl.] ¶¶55-56.

Thus, neither Takeda nor Ophir suggest storing shares generated from a dataset on two or more physical storage devices. Consequently, this claim requirement is a “missing limitation” that goes to the heart of the invention of claim 1. As the Federal Circuit has made clear, such a limitation cannot be supplied by conclusory statements or general conclusions about “basic knowledge” or “common sense” like those in the Petition (*see* Pet., 31). *Arendi S.A.R.L. v. Apple, Inc.*, 832 F.3d 1355, 1362-63 (Fed. Cir. 2016); *K/S HIMPP v. Hear-Wear Techs., LLC*, 751 F.3d 1362, 1366 (Fed. Cir. 2014) (citation omitted); *DSS Tech. Mgmt. v. Apple Inc.*, 885 F.3d 1367, 1374-75 (Fed. Cir. 2018) (“common sense” can supply a missing limitation “only when ‘the limitation in question was

unusually simple and the technology particularly straightforward.””).

Here, the limitation is of crucial importance and is neither unusually simple nor particularly straightforward. The ’140 teaches that dividing a dataset into shares and storing those shares across multiple storage devices significantly enhances both the security and recovery of stored data. *See* Ex. 2003 [Rubin-Decl.] ¶44 (discussing Ex. 1001 [’140] 95:15-18, 21:4-12, 69:16-22, 53:6-18, 60:43-51). Similarly, the field of data security is complicated and intricate. Ex. 2003 [Rubin-Decl.] ¶58.

Petitioner thus fails to demonstrate that either reference discloses or suggests storing the plurality of different blocks of a single dataset disclosed in Takeda across the plurality of physical storage devices disclosed in Ophir. Petitioner also fails to demonstrate that a POSITA would be motivated and able to modify the combination so as to satisfy this missing limitation by anything other than hindsight recourse to the ’140’s insight. As Dr. Rubin explains:

Petitioner fails to demonstrate that a POSITA would be motivated to combine Ophir and Takeda such that Takeda’s blocks are stored amongst multiple storage devices. Petitioner posits implementing Takeda’s block-based encryption scheme in conjunction with Ophir’s mediator. *See* Pet., 17 (“A POSA would have been motivated by Takeda to implement Takeda’s ciphering/deciphering as the encryption/decryption performed by

Ophir's encryption/decryption unit (*e.g.*, encryption/decryption unit 517) to achieve 'a more reliable ciphering technique' as taught by Takeda."). Petitioner contends that, "[b]esides modifying Ophir's encryption/decryption techniques to use those of Takeda, Ophir-Takeda need not change Ophir's other aspects." *Id.*, 25.

Thus, Petitioner's combination is focused upon implementing Takeda's block-based scheme in Ophir's system. In other words, Petitioner contends that a POSITA would replace Ophir's encryption scheme, in which a file is encrypted and then stored on one of several available storage devices, with Takeda's scheme, in which the blocks of a file are separately encrypted and stored together in a single file on a storage device.

What is missing is any indication or argument that, even with that replacement, a POSITA would be motivated to divide Takeda's blocks from a given dataset across two or more of Ophir's physical storage devices. That was the inventors' insight, and Petitioner's argument to the contrary is pure hindsight.

Petitioner does not account for this gap as it merely asserts that the "Ophir-Takeda" combination "stores the plurality of shares ... on the system's plurality of storage devices." Pet., 31. Nor does Petitioner provide any reason for why importing Takeda's ciphering/deciphering scheme into Ophir would naturally result in Takeda's blocks from a given dataset being split up amongst multiple storage devices, as nothing in either reference suggests splitting a

dataset and storing different parts of that dataset amongst multiple storage devices.

Ex. 2003 [Rubin-Decl.] ¶¶59-62.

Petitioner, for its part, merely asserts, as a conclusion, that its combination will satisfy the missing limitation. Petitioner and its expert do not offer any reason why a POSITA would combine Ophir-Takeda such that this dispersed storing would be implemented, nor do they demonstrate that such storing would be a natural by-product of the combination. See Pet., 31 (“Ophir-Takeda stores the plurality of shares ... on the system’s plurality of storage devices” because “Takeda’s ciphering is performed in Ophir’s ‘Encryption/Decryption Unit’ 517 ... and the encrypted shares are stored on the above-discussed plurality of storage devices.”). The conclusory and incomplete analysis provided by Petitioner and its expert (*compare* Pet., 31 *with* Ex. 1003 [Bhattacharjee] ¶111), is legally insufficient to establish obviousness. *In re Magnum Oil*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements.”); 37 C.F.R. § 42.65; *Xerox Corp. v. Bytemark, Inc.*, IPR2022-00624, Paper 9, 15 (Aug. 24, 2022) (precedential) (“[T]he cited declaration testimony is conclusory and unsupported, adds little to the conclusory assertion for which it is offered to support, and is entitled to little weight.”); *Virtek Vision Int’l. ULC, v. Assembly Guidance Systems, Inc.*, __ F.4th

___, Case No. 22-1998 (“Virtek”), slip op. at 7, n.2 (Fed. Cir. Mar. 27, 2024) (declarant’s “conclusory assertions ... do not provide substantial evidence for finding a motivation to combine.”).

Petitioner’s mere identification of two references, one with multiple alleged data shares (Takeda) and one with alleged multiple storage devices (Ophir) is not sufficient to render their combination obvious as a matter of common sense or law. That dividing a dataset into blocks may have been known, and having a system with multiple storage devices was known, does not remotely establish that it was obvious to divide a dataset into shares and store those shares across physically distinct storage devices. In accord, it is well-settled that obviousness is not established merely because the “elements of the claimed invention were independently known in the prior art.” *Metalcraft of Mayville, Inc. v. Toro Co.*, 848 F.3d 1358, 1367 (Fed. Cir. 2017); *see also Virtek*, slip op. at 5 (“It does not suffice to meet the motivation to combine requirement to recognize that two alternative arrangements ... were both known in the art.”). Rather, an *ex post facto* obviousness assessment that “break[s] an invention into its component parts, then find[s] a prior art reference corresponding to each component” “would import hindsight into the obviousness determination by using the invention as a roadmap to find its prior art components” and “would discount the value of combining various existing features or principles in a new way to achieve a new result—often

the essence of the invention.” *Princeton Biochemicals, Inc. v. Beckman Coulter, Inc.*, 411 F.3d 1332, 1337 (Fed. Cir. 2005); *see also Ruiz v. A.B. Chance Co.*, 357 F.3d 1270, 1275 (Fed. Cir. 2004) (“This form of hindsight reasoning, using the invention as a roadmap to find its prior art components, would discount the value of combining various existing features or principles in a new way to achieve a new result - often the very definition of invention”).

Critically, what is required, and what Petitioner fails to provide, is proof that “one of skill in the art would have combined these particular references to produce the claimed invention.” *Metalcraft*, 848 F.3d at 1367. Where, as here, Petitioner fails to provide that proof, “we are left with only hindsight bias that *KSR* warns against.” *Id.* Petitioner’s naked assertion that its combination would satisfy the missing limitation is precisely the sort of conclusory, hindsight-driven analysis, the law forbids.

Thus, Petitioner has failed to show Ophir-Takeda discloses and/or render obvious limitations 1[a] and 1[b][2].

C. A POSITA Would Not Be Motivated To Combine Ophir-Takeda.

1. Petitioner’s Proposed Modification Of Ophir Based On Takeda Directly Conflicts With, And Undermines The Goal And Method Of Operation Of, Ophir.

As discussed, Petitioner proposes implementing Takeda’s block-based ciphering/deciphering techniques in Ophir’s system. Pet., 21. Petitioner, however,

cannot show that a POSITA would be motivated to modify Ophir in this fashion, because such a modification would be contrary to Ophir's goals and principle of operation.

Ophir is directed towards achieving a network that can securely store datasets from multiple clients, **without** requiring anything from the client to secure the dataset. See Ex. 1007 [Ophir] ¶8 (Ophir's goal is to provide "a network system for secure data storage ... which ... places no encryption burdens on the data clients"); *id.*, ¶¶5, 7 (similar). But adding Takeda's block encryption system would do the opposite, by requiring the data clients to be intimately involved in Ophir's data encryption. As Dr. Rubin explains:

Ophir is directed towards a networked data security system which can safely store data for multiple clients without putting any burdens on those clients relating to securing the data. Ophir expressly seeks to improve upon prior art data systems that "require[d] data clients to perform procedures related to critical cryptographic operations necessary for data security," which "puts stringent limitations on the ability of the system to optimize encryption methods and keys." Ex. 1007 [Ophir] ¶7. Ophir, thus, explains that there is "a need for, and it would be highly advantageous to have, a network system for secure data storage which offers both the flexibility, expandability, and scalability of the network, but **which also places no encryption burdens on the data clients.**" *Id.*, ¶8.

Ophir further explains that this “goal is met by the present invention.”
Id.

Consistent with this objective, Ophir’s “Summary of the Invention” explains that “[i]t is an objective of the present invention to provide secure data storage accessible to data clients over a network ***without requiring the data clients to perform any operations related to the security of the stored data***, including, but not limited to, encryption, decryption, key management, key distribution, key storage, and key updating.” *Id.*, ¶9. The system “imposes no requirement for data clients to perform security-related operations,” and any such involvement is purely optional. *Id.*

To carry out this objective, Ophir discloses a “secure data storage mediator,” which is not a user, but, rather, an automated entity that acts as a “central coordinator for data storage” (Ex. 1007 [Ophir] ¶13), that facilitates the encryption/decryption such that “data clients do not have to be involved in any aspects of the encryption/decryption of data in storage.” *Id.*, ¶15. Ophir confirms that “[t]he novel aspects of the present invention lie in the protection of data for storage, which the mediator performs over the network without imposing ***any*** compulsory involvement of the data clients.” *Id.*; *see also id.*, ¶28 (“In the case of the prior-art ***requirement for data clients to participate*** in the encryption and/or decryption processes, the ***lack of such a requirement*** by the present invention is a ***clear-cut advantage***.”).

Thus, the problem Ophir is trying to solve, its goal, and the novel aspect of its system is to have a networked data storage system that “places no encryption burdens on the data clients” (*id.*, ¶8), does not “requir[e] the data clients to perform any operations related to the security of the stored data” (*id.*, ¶9), and ensures that data is securely stored “without imposing **any** compulsory involvement of the data clients” (*id.*, ¶15).

Ex. 2003 [Rubin-Decl.] ¶¶67-70.

Incorporating Takeda’s ciphering/deciphering techniques in Ophir’s system, as Petitioner proposes (Pet., 21), however, would thwart Ophir’s goal and principle of operation. As Dr. Rubin explains:

Adding Takeda’s ciphering/deciphering techniques to Ophir’s system would be entirely contrary to Ophir’s goal and principle of operation because Takeda’s methodology **requires** the data clients (Takeda’s “users”) to be heavily involved in the ciphering/deciphering processes.

Takeda’s ciphering/deciphering processes requires user involvement at multiple stages, including providing or determining a ciphering attribute (*e.g.*, a “ciphering key[]” or “algorithm[] for ciphering” per Takeda, ¶42) for each block into which the data set is divided, and when/whether to continue or end the ciphering/deciphering processes. Takeda makes clear that the user determines, inputs, and sets the ciphering attribute, and does not disclose any alternative to the user’s involvement. Ex. 1008 [Takeda]

¶9 (“*a user* appropriately *determines the ciphering attribute*”), ¶56 (“*a user* who wishes to cipher the plaintext *inputs the attribute*”), ¶88 (“the *user sets the attribute* for each block”), Fig. 19 (illustrating user input in ciphering). Takeda’s user also determines whether the ciphering/deciphering processes should continue or terminate. Ex. 1008 [Takeda] ¶56 (user must “input whether or not the ciphering processing is to be ended”), Fig. 5 (illustrating user input in ciphering process), ¶58 (“... the *user* ... instruct[s] whether or not to end the deciphering process.”), Figs. 5-6 (illustrating user input in deciphering process). Petitioner recognizes that the data clients are intimately involved in Takeda’s encryption operations, noting that “[t]he user sets the attribute for each block using attribute setting section 12...” Pet., 15.

Thus, contrary to Ophir’s goal, Takeda **requires** user involvement throughout the ciphering/deciphering process. While Takeda discusses a user (usually a person) rather than a data client (usually a computer), the user’s involvement would be manifested through computers (data clients) and further introduces the additional element of human error.

Thus, introducing Takeda’s encryption/decryption process in Ophir’s system would be entirely contrary to Ophir’s goals and principle of operation. Whereas Ophir’s entire purpose is to create a networked system that does not require any involvement by the data clients in the encryption/decryption process, particularly for data in storage, Takeda **requires** not only data clients, but the human users

thereto, to be intimately involved in key aspects of the ciphering/deciphering process of stored data.

Ex. 2003 [Rubin-Decl.] ¶¶72-75.

Significantly, Petitioner proposes adding Takeda's techniques to Ophir, **without** changing Takeda's techniques or otherwise modifying Ophir's system. Pet., 21-22 (substitution "would have predictably resulted in Ophir's data ciphering/deciphering ... being performed **as described by Takeda**"), 25 ("Besides modifying Ophir's encryption/decryption techniques to use those of **Takeda**, Ophir-Takeda **need not change Ophir's other aspects**."). Thus, Petitioner does not suggest that Takeda's ciphering/deciphering techniques would be modified to operate without the necessary involvement of the users/data clients.

Because Petitioner's proposed modification of Ophir in view of Takeda would undermine Ophir's goal and principle of operation, and there is further no suggestion in the references that they can or should be modified as proposed, it cannot support a finding of obviousness. *In re Ratti*, 270 F.2d 810, 813 (CCPA 1959) (proposed combination is improper where it would change basic principles of operation of one of the references); *Plas-Pak Indus., Inc. v. Sulzer Mixpac AG*, 600 F. App'x. 755, 759 (Fed. Cir. 2015) ("a change in a reference's 'principle of operation' is unlikely to motivate a person of ordinary skill to pursue a combination with that reference"); *Polaris Indus., Inc. v. Arctic Cat, Inc.*, 882 F.3d

1056, 1069-70 (Fed. Cir. 2018) (rejecting obviousness determination, finding that evidence combination “would run contrary to one of [the primary reference]’s stated purposes” necessarily weighed against “motivat[ion] to combine”).

Petitioner, moreover, never confronts the fact that its combination is at cross purposes with Ophir, which is itself sufficient reason to reject Petitioner’s combination. *Arctic Cat Inc. v. Polaris Indus.*, 795 Fed. App’x. 827, 833 (Fed. Cir. 2019) (“[t]he Board must weigh the benefits and drawbacks of the modification against each other, to determine whether there would be a motivation to combine.”); *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1331-32 (Fed. Cir. 2019) (similar). The Petition’s “disregard” of Ophir’s teachings and its reliance on modifications based on Takeda that are detrimental to Ophir’s goals while “identify[ing] no reason at all” to do so, are fatal to its contention that the claims are obvious in light of those references. *E.g., Institut Pasteur v. Focarino*, 738 F.3d 1337, 1345-46 (Fed. Cir. 2013).

2. The Petition Fails To Show That A POSITA Would Be Motivated To Modify Ophir To Store Blocks From A Single Dataset Amongst Multiple Storage Devices.

The Petition identifies three purported reasons why, it contends, a POSITA “would be motivated by Takeda to implement Takeda’s ciphering/deciphering technique (splitting the data into blocks and encrypting different blocks with different ciphering attributes) in Ophir.” Pet., 17-25. Even if such a motivation

existed, however, those reasons are insufficient to establish obviousness here. The reasons offered by Petitioner do not (and cannot) explain why or how a POSITA would modify Ophir to store blocks amongst multiple storage devices, or that a POSITA would have a reasonable expectation of success in doing so. *See* Section III.B. A combination that does not achieve the claimed invention cannot establish that the invention is obvious. *Medichem, S.A. v. Rolabo, S.L.*, 437 F.3d 1157, 1165 (Fed. Cir. 2006) (“an obviousness determination requires ... that a skilled artisan would have perceived a reasonable expectation of success in making the invention via that combination.”).

As Dr. Rubin explains:

Petitioner contends that a POSITA would modify Ophir by “simpl[y] substitut[ing]” Takeda’s block-based encryption/decryption technique for Ophir’s file-based encryption/decryption technique. Pet., 21. It explains that a POSITA would be motivated to “split[] the data into blocks and encrypt[] different blocks with different ciphering attributes.” *Id.*, 18. A POSITA would be motivated to do so, Petitioner asserts, “to achieve ‘a more reliable ciphering technique’” wherein ““even if a part of the data is deciphered, the rest of the data is not easily deciphered.”” *Id.*, 17 (citing Ex. 1008 [Takeda] ¶6).

But the “simple substitution” envisioned by Petitioner (and, as explained, it is neither simple nor even feasible), whatever the reason for it, would not achieve the claimed invention. Claim 1 requires, in

addition to the creation and encryption of shares, that different shares be stored amongst multiple storage devices. But even assuming that “blocks” in Takeda correspond to “shares” in claim 1, Takeda cannot motivate one of skill to modify Ophir to store different shares amongst multiple storage devices because it does not teach or contemplate anything other than storing all of the blocks together in a single file on a single storage device. *See* Ex. 1008 [Takeda] ¶¶48, 91-92, Figs. 1B, 2B, 3A, 3B.

Nor would a POSITA have a reasonable expectation of success in achieving the “different blocks on different storage devices” aspect of claim 1 by modifying Ophir in light of Takeda. Takeda does not even teach how a dataset should be divided into blocks, much less how blocks could be identified and tracked on different storage systems in order to permit reconstruction of a data set. *See* Ex. 1008 [Takeda] ¶64 (stating that “[i]n the above description ... a way of forming blocks ... has not particularly been described [but] may be designated by the user or by using a specific algorithm.”). Nor could it since all of Takeda’s blocks from a given dataset are stored together. And the gap is not filled by Ophir itself, because the Ophir mediator only stores files (Ex. 1007 [Ophir] ¶¶12-13), not parts of files, and it discloses no way to track, identify, find or retrieve different blocks stored on different storage devices, as would be required to reconstruct a dataset and provide it to a client device.

Ex. 2003 [Rubin-Decl.] ¶¶78-80.

The Petition never confronts, or even mentions, its failure of proof on this point. *See* Pet., 17-25. This is another independent reason why the Petition fails to establish the claims are obvious over Ophir/Takeda. *See* Section III.A-B, *supra*.

D. Ophir Does Not Disclose And/Or Render Obvious “A Directory Mapped To The Plurality Of Physical Storage Devices.”

Each challenged claim requires “a directory mapped to the plurality of physical storage devices” (limitation 1[b][1]). Ex. 1001 [’140] cl. 1. Petitioner argues that Ophir discloses and/or renders obvious this limitation through Ophir’s disclosure of: (1) a mediator (Pet., 32-35); (2) block services (Pet., 35-37), or (3) file services. Pet., 37-39. Petitioner’s arguments, addressed in turn below, fail. Indeed, Petitioner’s reliance on three different disclosures, and its inability to point to one particular aspect of Ophir as corresponding to this key limitation, underscores that Ophir simply does not disclose it.

1. Ophir’s Mediator Does Not Disclose Or Render Obvious The Required Directory.

Petitioner first identifies Ophir’s mediator as disclosing and/or rendering obvious a “directory mapped to the plurality of physical storage devices.”

Petitioner contends that:

Because the client can retrieve stored data through the mediator without knowing the data’s stored location ... POSAs would have understood that Ophir maintains a directory mapping where in the

physical storage locations each piece of client data is stored, so that the mediator can retrieve the data when the client desires.

Pet., 34 (citing Ex. 1003 [Bhattacharjee] ¶117).

Petitioner does not (and cannot) point to any express disclosure of a “directory” in Ophir, much less one “mapped to the plurality of physical storage devices,” because Ophir does not disclose how the mediator retrieves data. Ex. 1007 [Ophir] ¶13 (“the mediator is able to store data in, and retrieve data from, any suitable storage device having access to the network.”). And Petitioner does not even attempt to demonstrate that Ophir’s mediator would inherently use a “directory mapped to the plurality of physical storage devices” to retrieve requested data. *Trintec Indus., Inc. v. Top-U.S.A. Corp.*, 295 F.3d 1292, 1295 (Fed. Cir. 2002) (inherency requires missing element be “***necessarily present***,’ not merely probably or possibly present, in the prior art.”).

Petitioner alternatively asserts that “maintaining such a directory mapped to the physical storage devices would have been a conventional and obvious way to implement what Ophir describes to allow the mediator to retrieve client data for the data clients when the client does ‘not know the locations where the data is stored.’”

Pet., 34 (citing Ex. 1007 [Ophir] ¶13; Ex. 1003 [Bhattacharjee] ¶¶117-119).

Petitioner’s analysis, however, is conclusory and insufficient to establish obviousness. *Virtek*, slip op. at 5 (“It does not suffice to meet the motivation to

combine requirement to recognize that two alternative arrangements ... were both known in the art.”). The Petition attempts to bolster this conclusion with testimony from Dr. Bhattacharjee regarding Exhibits 1014 and 1016. *See* Ex. 1003 [Bhattacharjee] ¶¶118-119.³ But even if his testimony on these exhibits could be considered, the references themselves are not a part of any ground set out in the Petition and thus cannot provide a basis for modifying Ophir-Takeda. *Edwards Lifesciences Corp. v. Aortic Innovations LLC*, IPR2023-01232, Paper 10, 36 (Feb. 7, 2024) (“Petitioner’s rationale to modify Spenser II based on inchoate and ***unasserted combinations*** is also unpersuasive.”) (denying institution); *RPX Corp. v. Parity Networks, LLC*, IPR2018-00097, Paper 7, 14 (Apr. 24, 2018) (“Petitioner ... does not assert these references are part of the combination. ... Relying on knowledge in the art does not ... show a motivation to combine ...”) (denying institution); *Parsons Xtreme Golf, LLC v. Taylor Made Golf Co., Inc.*, IPR2018-00518, Paper 14, 18 n.4 (July 18, 2018) (similar); *see also Belden Inc. v. Berk-Tek*

³ The substance of Dr. Bhattacharjee’s testimony regarding these exhibits is not presented in the Petition, and thus should not be considered for this additional reason. 37 C.F.R. § 42.6(a)(3); *3M Co. v. Evergreen Adhesives, Inc.*, 860 Fed. App’x. 724, 725-728 (Fed. Cir. 2021) (Board properly declined to consider improperly incorporated testimony).

LLC, 805 F.3d 1064, 1073 (Fed. Cir. 2015) (“[O]bviousness concerns whether a skilled artisan not only *could have made* but *would have been motivated to make* the combinations or modifications of prior art to arrive at the claimed invention.”).

Thus, Petitioner has failed to show Ophir’s mediator discloses or renders obvious “a directory mapped to the plurality of physical storage devices.”

2. Ophir Does Not Disclose The Required Directory By Disclosing “Block Services.”

Petitioner alternatively argues that Ophir’s mediator provides “block services” “which POSAs would have understood means that the mediator provides block-based virtual storage.” Pet., 35. Petitioner further contends that “POSAs would have understood that storage blocks presented to a data client are blocks of a virtual (rather than physical) disk, wherein the virtual disk comprises a directory mapped to the plurality of physical storage devices.” Pet., 36 (citing Ex. 1016, 5:39-58; Ex. 1017, ¶20; Ex. 1018, ¶29, Fig. 16; Ex. 1003 [Bhattacharjee] ¶132).

As an initial matter, Petitioner quotes Ophir for the proposition that the mediator provides “virtual secure storage,” suggesting that this corresponds to the “virtual disk” of the ’140’s claim 1. Pet., 9. It does not; Ophir explicitly states that its storage is considered “virtual” not because it is presented as a single storage device, but only “because the data ... can be stored on a variety of storage devices using a variety of protocols, technologies and services ...” Ex. 1007 [Ophir] ¶33.

Petitioner does not (and cannot) point to any disclosure in Ophir of a “directory” in connection with the disclosed “block services,” much less one “mapped to a plurality of physical storage devices.” Petitioner’s argument that a POSITA would “understand” that block services entail such a directory requires that it show that such a directory is inherent, but it does not even attempt to prove that Ophir’s system *necessarily* includes a directory mapped to a plurality of physical storage devices. See Section III.D.1 (citing *Trintec*).⁴

Petitioner alternatively argues a mapping directory “would have been the conventional and obvious way to implement what Ophir describes” because “the mediator provides ‘virtual’ storage, data is accessed by a client on a block level, and the client need not know where the data is physically stored.” Pet., 36 (citing Ex. 1007 [Ophir] ¶¶13, 33; Ex. 1019 1:39-50, 2:13-24; Ex. 1015, Fig. 14A, 7:32-40, 8:55-9:3; Ex. 1016, 2:57-3:3, 5:39-58; Ex. 1017 ¶20; Ex. 1018 ¶¶29, 45-48; Ex.

⁴ Confirming that Ophir does not disclose a “directory mapped to a plurality of physical storage devices,” Petitioner provides unexplained citations to ancillary references (Exs. 1016, 1017, and 1018). Pet., 36. These references have no bearing upon what *Ophir’s* system requires and no explanation is given for why these references would necessitate a mapped directory to facilitate Ophir’s “block services.”

1003 [Bhattacharjee] ¶133). Thus, here too, Petitioner’s argument is rooted in ancillary references (Exs. 1015-1019) that are not a part of Petitioner’s combination and, as such, cannot provide a motivation to modify Ophir. See Section III.D.1, *supra* (citing *Edwards*, IPR2023-01232, Paper 10 at 36; *RPX*, IPR2018-00097, Paper 7 at 14; *Parsons*, IPR2018-00518, Paper 14 at 18 n.4).

But even if these references were considered, they do not demonstrate that a mapped directory was a conventional and obvious solution. As Dr. Rubin explains:

Petitioner cites to specific excerpts of Exhibits 1015-1019 in support of its proposition that “implementing a virtual disk by using a mapping directory ... would have been the conventional and obvious way to implement what Ophir describes.” But Petitioner fails to show its citations actually teach a “directory mapped to the plurality of physical storage devices.” Some of Petitioner’s citations discuss some form of mapping, but not a directory (Ex. 1017 ¶20; Ex. 1018 ¶¶29, 45-48), another discusses a “directory” but not mapping (Ex. 1016, 2:57-3:3, 5:39-58), while the remaining two discuss neither (Ex. 1019 1:39-50, 2:13-24; Ex. 1015, Fig. 14A, 7:32-40, 8:55-9:3). Nor does Petitioner expand on how these five separate references should be taken together to modify Ophir such that it would teach this limitation. Petitioner simply has not shown that a directory mapped to a plurality of physical storage devices is a conventional and obvious way of implementing Ophir.

Ex. 2003 [Rubin-Decl.] ¶91.

Petitioner also fails to show any motivation to modify Ophir in the face of these five references. As discussed, Ophir is already capable of recovering data without disclosing a mapped directory. Ex. 1007 [Ophir] ¶13; Section III.D.1. Petitioner offers no motivation to modify Ophir when it already performs the recovery task in a different way. *Virtek*, slip. op. at 6 (“The mere fact that these possible arrangements existed in the prior art does not provide a reason that a skilled artisan would have substituted the one-camera angular direction system in Kietler and Bridges with the two-camera 3D coordinate system disclosed in Briggs.”).

Petitioner further contends that implementing a “directory mapped to the plurality of physical storage devices” would be appropriate because “virtual disks were known to use a mapping directory to map virtual storage blocks to physical storage locations.” Pet., 37 (citing Ex. 1007 [Ophir] ¶¶13, 33; Ex. 1003 [Bhattacharjee] ¶¶133-34). But, as discussed, Ophir does not disclose a “virtual disk” in the way the term is used in claim 1. And Petitioner’s citation to Ophir’s

¶¶13 and 33 do not help it, because neither paragraph discloses a mapped directory.⁵

3. Ophir Does Not Disclose The Required Directory By Disclosing “File Services.”

Petitioner also contends that Ophir’s disclosure of “file services” discloses and/or renders obvious the directory limitation 1[b][1]. Petitioner claims that because a client can retrieve files from storage via the mediator, a “directory mapped to the plurality of physical storage devices” must be included as part of Ophir’s “file services.” See Pet., 38 (citing Ex. 1007 [Ophir] ¶¶46-47; Ex. 1003 [Bhattacharjee] ¶¶137-40). But Petitioner points to no express disclosure in Ophir for this proposition and again, invoking inherency, relies upon what a POSITA supposedly would have “understood” because “that is how file systems were known to work.” Pet., 38. Petitioner cites to Ophir at ¶¶46-47 for such knowledge, but these citations do not mention a directory or the required mapping, and certainly do not demonstrate that a directory mapped to the plurality of storage

⁵ Petitioner’s only other citation is to Dr. Bhattacharjee’s testimony at ¶¶133-34 which bootstraps further discussion not presented in the Petition. Thus, this incorporated by reference discussion should not be considered.

devices is ***necessarily*** present in Ophir.⁶ See Section III.D.2, *supra* (citing *Trintec*; *Microsoft Corp.*; *LBT IP*).

Petitioner alternatively argues “maintaining a ‘directory’ mapped to the physical storage locations would have been a conventional and obvious way to implement Ophir’s file system because a file system conventionally maintains such directories.” Pet., 39. But Petitioner fails to explain why a POSITA would modify Ophir to have its “file services” include a “directory mapped to the plurality of physical storage devices” when it already is able to retrieve stored information. See Pet., 39; Ex. 1007 [Ophir] ¶13. Nor does Petitioner discuss what benefits the modification would render, what changes must be made to Ophir’s system, or what success a POSITA would have making such a modification. *Id.* Such a cursory and incomplete analysis cannot establish obviousness. *Virtek*, slip. op., 5 (“It does not suffice to meet the motivation to combine requirement to recognize that two alternative arrangements ... were both known in the art.”).

⁶ Dr. Bhattacharjee otherwise attempts to bolster his inherency conclusion by discussing Exhibits 1020 and 1021. See Ex. 1003 [Bhattacharjee] ¶¶139-40. But the Petition does not substantively discuss these Exhibits, and thus the Board should not consider Dr. Bhattacharjee’s improperly incorporated discussion. See Section III.D.1 (citing authorities).

IV. DICKINSON-TAKEDA DOES NOT DISCLOSE OR RENDER OBVIOUS THE CLAIMS OF THE '140 PATENT (ALL CLAIMS, GROUND 4).

A. The Claims Are Directed To Different Technology Than Dickinson.

1. The Trust Engine Of Dickinson Is A Fundamentally Different Invention Than The Virtual Disk Of Claim 1.

Petitioner's fourth ground primarily relies upon Dickinson (Ex. 1005).

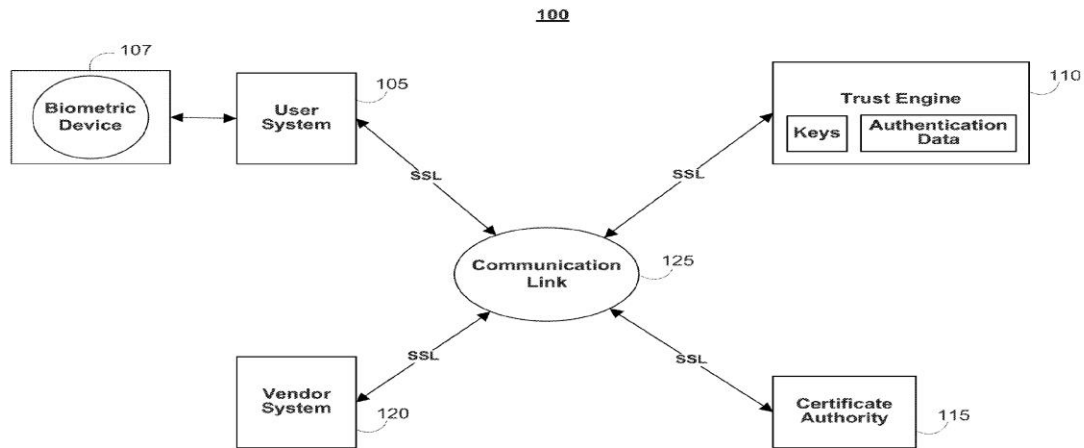
While it has certain inventors in common with the '140 (Orsini, O'Hare and Davenport), Dickinson is directed to one of their earlier and much different inventions, namely, a "trust engine" for "facilitating an authentication relating to an electronic transaction between a first and a second user." Ex. 1005 [Dickinson] Abstract.

As Dr. Rubin explains:

Dickinson is directed to a "trust engine" for "accessing and storing sensitive information, such as user authentication data and public and private encryption keys" that can be used to "uniquely identify a user" of the system. Ex. 1005 [Dickinson] 10:20-23. Examples of data accessible using the trust engine include "a user identification number, one or more biometrics, [or] a series of questions and answers ... answered initially by the user at enrollment." *Id.*, 10:23-25. The trust engine compares the authentication data provided by the user in "a current transaction[]" to the authentication data provided at an earlier time, such as ... during

enrollment.” *Id.*, 10:27-29. “Once the user produces the appropriate [current] authentication data and the trust engine 110 determines a positive match” with the stored data “provided at the time of enrollment (enrollment authentication data), the trust engine 110 provides” functionality to securely complete the current transaction, such as “hashing, digitally signing, encrypting and decrypting, ... distributing digital certificates, and the like.” *Id.*, 11:3-9.

Notably, in contrast to the claimed inventions, Dickinson’s trust engine does not present a virtual disk of any sort to the user. In fact, the opposite is true: the user provides authentication data to the trust engine, data that can be compared to the authentication data already securely stored within the trust engine. For example, a user may provide a fingerprint to verify his or her identity to facilitate the issuance of a digital certificate by a certificate authority. The trust engine receives digitized fingerprint data from reader and compares it to fingerprint data for that user previously stored within the trust engine during enrollment. If it finds a match, the trust engine may transmit a request to the certificate authority. Dickinson’s trust engine is shown in relation to the user and the certificate authority in Figure 1:



Id., Fig. 1.

Dickinson's trust engine is comprised of several modules, as shown in its Figure 2, including a transaction engine for communicating through a communications link with users and third parties. The modules also comprise (1) a depository for storing enrollment authentication data and cryptographic key data, (2) an authentication engine for comparing enrollment authentication data provided by the depository to current authentication data, and (3) a cryptographic engine to provide required cryptographic services if the user is authenticated. *See generally id.*, 13:5-15:33. Notably, as shown in Figure 2, the transaction engine is the only module that communicates with users outside the trust engine.

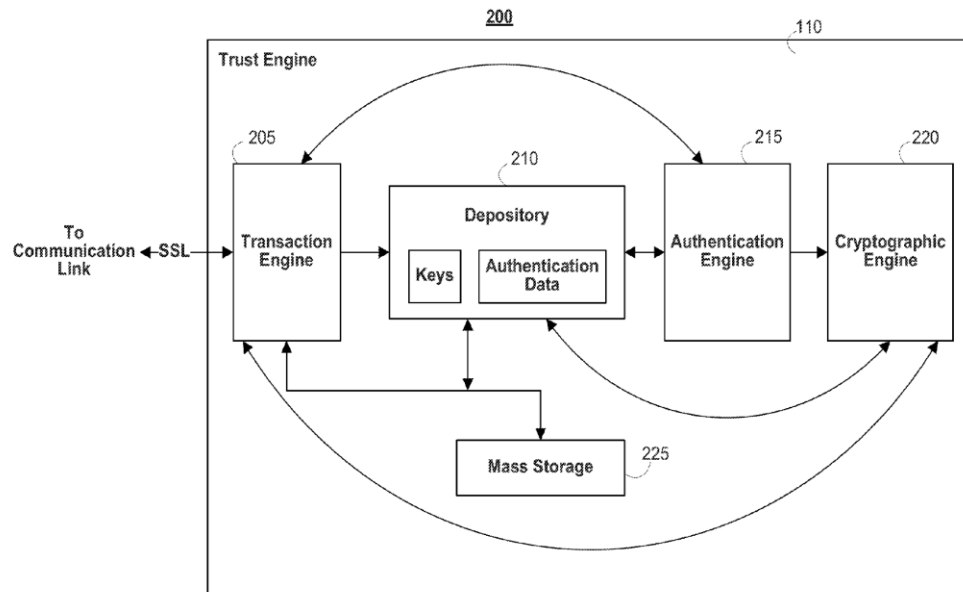


FIG. 2

Id., Fig. 2.

The depository 210 “comprises one or more lightweight directory access protocol (LDAP) servers” used to store enrollment authentication data. *Id.*, 16:25-29. Depository 210, shown in Figure 4 below, may then comprise “a single logical memory structure indexing authentication data ... to a unique user ID.” *Id.*, 16:28-31, 17:4-5.⁷ As shown on the right side of Figure 4, the depository

⁷ Petitioner misleadingly suggests that Dickinson equates its “single logical memory structure indexing authentication data ... to a unique user ID” to a “virtual disk.” Pet., 71-72. However, there is no such disclosure in Dickinson—in fact, the term “virtual” is not used in the reference.

provides data to the trust engine's mass storage, to the authentication engine, and to its cryptographic engine, but does not provide data outside the trust engine.

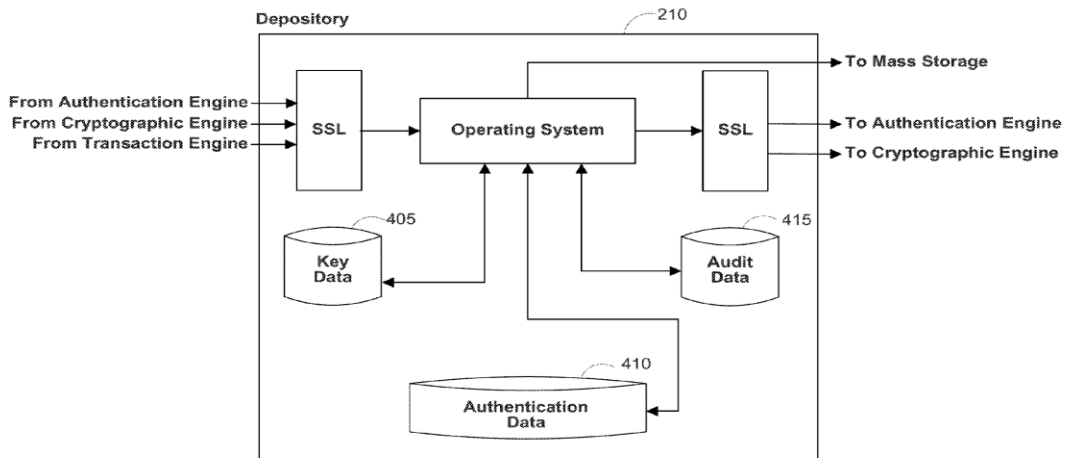


FIG. 4

Id., Fig. 4.

As shown in Figure 7 below, Dickinson's depository may comprise "distinct and physically separated data storage facilities," shown as D1-D4. *Id.*, 17:4-5. Each data storage facility "may ... comprise some or all of the elements disclosed with reference to depository 210 of Figure 4." *Id.*, 19:19-20.

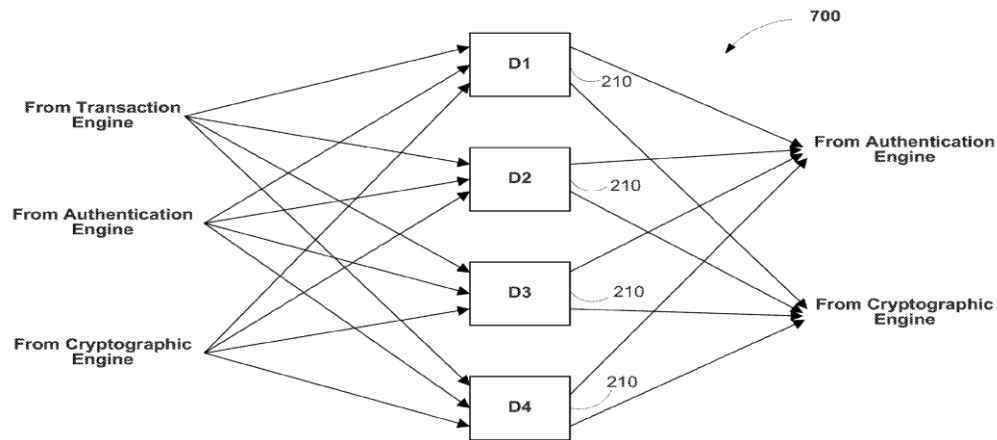


FIG. 7

Id., Fig. 7.

In one embodiment, the authentication engine includes a “splitting module” to divide enrollment authentication data into one or more “portions” or shares, and different shares of that data may be stored in different storage facilities to promote data security, availability and redundancy. *Id.*, 19:31-34, 20:1-14.

Ex. 2003 [Rubin-Decl.] ¶¶102-107.

In short, Dickinson’s trust engine is very different from the virtual disk of claim 1. Even if it is argued that both systems may divide data into a plurality of shares for storage across a plurality of storage devices, Dickinson does not present, among other things, a virtual disk to a client system as required by claim 1 of the ’140. Specifically, there is no disclosure in Dickinson that any information about the “logical memory structure” of the trust engine’s “depository,” which Petitioner contends corresponds to the “virtual disk” of claim 1, is presented to the trust

engine's users. Nor is there any disclosure regarding what that logical memory structure comprises, beyond authentication and cryptographic data, or what it may be mapped to. As shown below, Dickinson does not disclose key features of the claimed inventions.

2. The '140 Patent Adds Critical New Material Regarding The Limitations Of Claim 1 That Is Not Disclosed In Dickinson.

a. *New Disclosures In The '140 Patent Not Disclosed In Dickinson Teach Limitation 1[b][1].*

Dickinson's figures and disclosure are largely present in the '140's specification. But the '140 has significant new additions that are not present in Dickinson and disclose the inventions at issue here. The '140's specification adds **40 new figures and over forty-five columns of new material** not present in Dickinson. *Compare* Ex. 1005 [Dickinson] *with* Ex. 1001 ['140] Figs. 21-60, 1:1-2:26, 2:30-58, 3:44-6:45, 7:41-9:12, 10:40-41, 10:63-65, 18:56, 19:12-15, 20:20-22, 20:38-39, 20:48-50, 21:1-3, 21:24-40, 22:13-21, 30:53-55, 34:7-12, 36:60-64, 37:36-40, 44:42-44, 52:55-99:23.

Much of this new content directly relates to claim 1's "virtual disk" limitation. As Dr. Rubin explains:

Limitation 1[b][1], again, requires "a secure storage system configured to" "present to a client device a virtual disk, the virtual disk comprising a directory mapped to the plurality of physical storage devices such that physical locations of the shares are hidden

from the client device.” Limitation 1[b][1] is disclosed in the ’140, but not in Dickinson.

The ’140, but not Dickinson, introduces a “Parsed File System (PFS)” which, as will be shown, is an example of limitation 1[b][1]:

A Parsed File System (PFS) *is a distributed file system which incorporates a directory*, shared among file servers, which is itself parsed. *The PFS directory is the clients’ view of all file assets*. The set of files visible to an individual client is controlled by that client’s group keys. *The PFS may maintain a directory that contains information about the locations of the shares (for example, the file server location) and the names of the files at the share locations that include the share data.*
...

The underlying distribution of shares may be hidden from users/clients by the PFS such that files appear as a file storage system based on the PFS directory. The PFS provides an interface that allows users to access parsed files without knowledge of the location of shares.

Id., 87:16-43.

Thus, in accordance with limitation 1[b][1], the PFS is, *e.g.*, “a distributed file system which incorporates a directory.” *Id.*, 87:16-18. The “PFS directory is the clients’ view of all file assets” and, thus, is “present[ed] to a client device.” *Id.*, 87:18-19. The PFS directory is

“mapped to the plurality of physical storage devices,” as it may “contain[] information about the locations of the shares (for example, the file server location) and the names of the files at the share locations that include the share data.” *Id.*, 87:20-24. The PFS directory, moreover, “provides an interface that allows users to access parsed files ***without knowledge of the locations of shares***,” showing the directory may be mapped “such that physical locations of the shares are hidden from the client device.” *Id.*, 87:37-39.

...

The PFS directory is disclosed in the ’140 but not in Dickinson. *Compare* Ex. 1001 [’140] 87:16-88:56 *with generally* Ex. 1005 [Dickinson]. Petitioner and Dr. Bhattacharjee further ignore the ’140’s teachings relating to the PFS directory. ...

Ex. 2003 [Rubin-Decl.] ¶¶110-114.

b. *New Disclosures In The ’140 Patent Not Disclosed In Dickinson Teach Limitation 1[B][3].*

Limitation 1[b][3] requires “includ[ing] with each of the plurality of shares data indicative of ***the at least one session key*** used to secure the dataset” (1[b][3]).

Dickinson entirely lacks this disclosure, as Dr. Rubin explains:

Dickinson introduces and discusses a “session key,” but only in connection with Figure 12. *See, e.g.*, Ex. 1005 [Dickinson] 32:26-31; Fig. 12. But nowhere in Dickinson’s discussion of Figure 12 is a “plurality of shares” even mentioned (1[a]) much less “includ[ing] with the plurality of shares data indicative of the at least one session

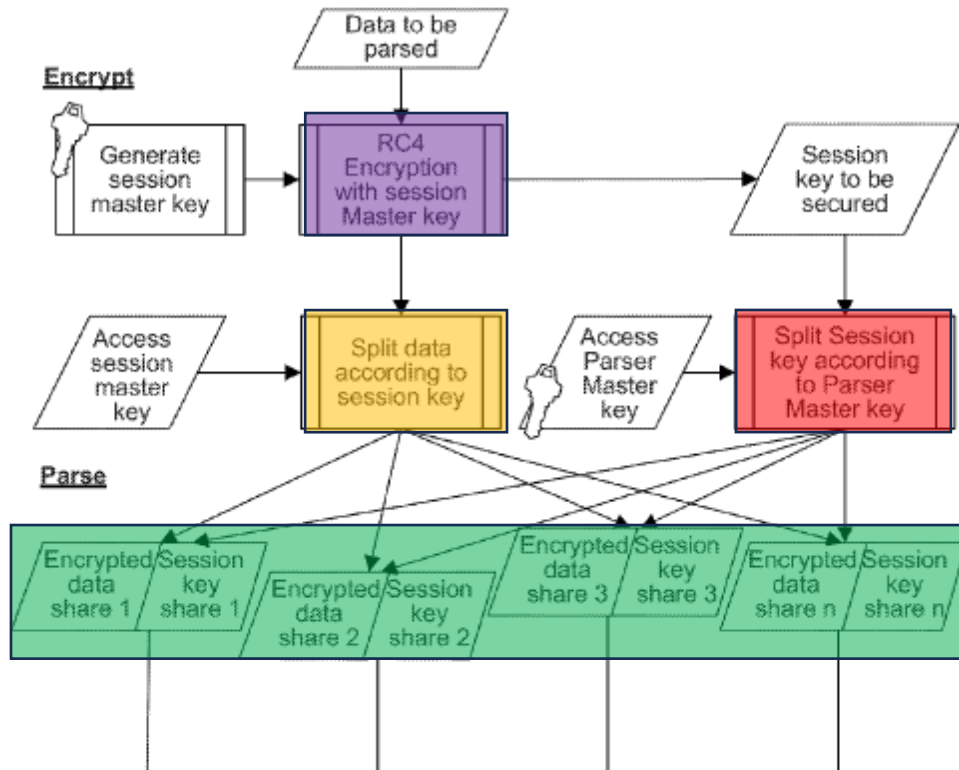
key” as required by limitation 1[b][3]. See Ex. 1005 [Dickinson] 32:26-31.

In any event, Petitioner does not even rely on Dickinson’s teachings relating to a “session key” to satisfy limitation 1[b][3], or limitation 1[a] (also concerning a “session key”) for that matter. Petitioner instead relies upon Dickinson’s teachings relating to Figure 8, which do not mention or concern session keys at all. See Pet., 69-70.

Indeed, substantial additions to the ’140 over Dickinson focus upon the handling of “session keys.” Indeed, session keys are mentioned 108 times in the ’140 patent, and 93 of these instances are only in the ’140. Session keys are, moreover, a feature in eight new figures (Figs. 21-24, 33-35, 39) and are further discussed in connection with eight more figures. *Id.*, 71:12-21 (Fig. 30), 76:4-49 (Fig. 36), 76:62-77:4 (Fig. 37), 77:5-33 (Fig. 38), 78:20-40 (Fig. 40), 83:31-50 (Fig. 46), 85:33-53 (Fig. 46), 87:3-15 (Fig. 47). Thus, there is an enormous expansion on the teaching of a “session key” in the ’140, as compared to Dickinson.

As just one example, new Figure 21 teaches **encrypting a dataset with a “session master key,”** separating the encrypted data into four shares according to the pattern of the session master key, **separating the session master key according to the parser master key** and **appending the key data to the encrypted parsed data, wherein “[t]he resulting four shares of data will contain encrypted portions of**

the original data and portions of the session master key.” Ex. 1001
[’140] 56:19-36.



Ex. 1001 [’140] Fig. 21 (annotated).

Thus, among other new, relevant disclosures in the ’140, Figure 21 in the ’140 teaches “a plurality of shares” “associated with at least one session key” (1[a]) and storing “data indicative of the at least one session key with each of the plurality of shares” (1[b][3]). There is no corresponding teaching in Dickinson.

Ex. 2003 [Rubin-Decl.] ¶¶116-120.

B. Dickinson Was Expressly Considered By The Examiner During Prosecution And Should Not Be Revisited.

The Board may deny institution where “the same or substantially the same prior art or arguments previously were presented to the Office.” 35 U.S.C. § 325(d). The Board should exercise that discretion here because Dickinson, and substantially the same arguments about Dickinson, were considered and rejected by the Examiner.

In evaluating whether institution should be denied under § 325(d), the Board first determines whether the same or substantially the same art or arguments were presented to the Office. *Advanced Bionics, LLC v. MED-EL Elektromedizinische Geräte GmbH*, IPR2019-01469, Paper 6, 8, 10 (Feb. 13, 2020) (precedential). Here it is indisputable that Dickinson, and Petitioner’s arguments about Dickinson, were already considered by the Examiner.

First, the challenged claims require a “secure storage system configured to” “present to a client device a virtual disk ... comprising a directory mapped to the plurality of physical storage devices such that physical locations of the shares are hidden from the client device” (limitation 1[b][1]). Petitioner contends that Dickinson discloses or renders obvious the “virtual disk” limitation. Pet., 71-74.

The Examiner, however, considered this very argument and reached precisely the opposite conclusion during prosecution. In a June 25, 2015 Office

Action, the Examiner determined that “Dickinson...**does not explicitly disclose the limitation** such that the virtual disk comprising the directory mapping to the plurality of storage device [sic].” See Ex. 1002 [Pros.-Hist.] 1239. The Examiner was right. Critically, in reaching this conclusion, the Examiner was aware of (and, indeed, cited) the very portion of Dickinson Petitioner now contends discloses the “virtual disk” limitation. Ex. 1002 [Pros.-Hist.] 1237-1238 (citing Ex. 1005 [Dickinson] 16:28-31). Moreover, the Examiner was fully aware of, and quoted, Dickinson at 16:28-31 in considering whether this disclosure satisfied the limitation-in-question. Compare *id.*, 1236 (Examiner discussing Ex. 1005 [Dickinson] Fig. 7 and 17:4-5) with Pet., 71 (discussing Ex. 1005 [Dickinson] Fig. 7 and 17:4-5); see also Ex. 1002 [Pros.-Hist.] 1237-1239 (Examiner quoting Ex. 1005 [Dickinson] 16:28-31). Thus, the Examiner considered and rejected the very argument Petitioner relies upon here.

Second, the “session key” limitation (1[b][3]), which requires “includ[ing] with each of the plurality of shares data indicative of **the at least one session key** used to secure the dataset” was specifically added during prosecution to overcome Dickinson. In adding the limitation, the Applicant made it perfectly clear that Dickinson—Applicant’s own patent application—did **not** practice limitation 1[b][3] and that the limitation was a patentable improvement over Dickinson:

Independent claim 14 has been amended to recite, *inter alia*, a plurality of physical storage devices storing thereon a plurality of shares, the plurality of shares being associated with at least one session key used to secure a dataset, and a secure storage system configured to generate the plurality of shares and ***include with each of the plurality of shares data indicative of the at least one session key used to secure the dataset.*** Independent claim 14 patentably improves upon the combination of Dickinson and Bendix because neither Dickinson nor Bendix teach or suggest storing any additional information with each of the data shares, much less information used to secure a dataset. For example, Dickinson describes a data splitting process 800 in which *transformed versions* of an *entire* set of data "S" are stored in multiple data storage facilities. *See, for example*, pp. 21-22 and FIG. 8 at step 805. However, Dickinson does not teach or suggest ***including with each of the plurality of shares data indicative of the at least one session key used to secure the dataset***, as claimed. Bendix also fails to teach or suggest this subject matter, and nowhere does the Examiner contend that it might.

Ex. 1002 [Pros.-Hist.] 1245-1246 (emphasis in original). The Examiner confirmed that "Applicant's arguments ... have been fully considered and are persuasive," and correctly allowed the claims. Ex. 1002 [Pros.-Hist.] 1360.

When the Board finds that the same art and arguments were presented during prosecution (as is true here), it then considers whether the petitioner has demonstrated a material error by the Office. *Advanced Bionics*, IPR2019-01469,

Paper 6, 8, 10 (precedential). If there is no showing of material error, denial of institution is warranted. *Id.* In making this assessment, “[i]f reasonable minds can disagree regarding the purported treatment of the art or arguments, it cannot be said that the Office erred in a manner material to patentability.” *Id.*, 9.

Petitioner has not demonstrated any material error by the Office. Petitioner contends that, “[t]o the extent SFI argued in prosecution” that Dickinson does not teach the session key limitation, “the Examiner erred in agreeing with that mischaracterization.” Pet., 75. But Petitioner does not demonstrate any “error,” much less “material error,” on the Examiner’s part in concluding that Dickinson does not disclose storing data indicative of a session key with each of the shares. And Petitioner has not, and cannot, demonstrate that reasonable minds could not differ regarding whether Dickinson discloses limitations 1[b][1] and 1[b][3]. See also Section IV.A.2(b) *supra*.

Thus, the Board should not revisit Petitioner’s failed arguments under Dickinson. But, even if these recycled arguments were to be considered, Petitioner’s arguments still fail for multiple reasons.

C. Dickinson-Takeda Does Not Disclose Or Render Obvious Limitation 1[B][1].

Limitation 1[b][1] requires a “secure storage system configured to” “present to a client device a virtual disk ... comprising a directory mapped to the plurality

of physical storage devices such that physical locations of the shares are hidden from the client device.” As shown below, Dickinson does not disclose or render obvious: (1) a “directory,” (2) “mapped to the plurality of physical storage devices,” (3) that the “physical locations of the shares are hidden from the client device,” or (4) “a secure storage system configured to ... present to the client device a virtual disk.”

1. Dickinson Does Not Disclose A “Directory.”

Petitioner contends that Dickinson’s depository 210’s logical memory structure comprises an “index” and that this “index is a directory.” Pet., 72-73 (citing Ex. 1005 [Dickinson] 16:28-31; Ex. 1003 [Bhattacharjee] ¶¶230-231).

Petitioner’s argument fails, however, for it does not prove that Dickinson even discloses an “index,” much less that any such “index” would be a “directory.” Petitioner and Dr. Bhattacharjee rely upon Dickinson’s disclosure of depository 210’s “single logical memory structure indexing authentication data and cryptographic key data to a unique user ID” for limitation 1[b][1]. Ex. 1005 [Dickinson] 16:30-31. Petitioner, however, has not shown that this “logical memory structure” is an “index,” includes an “index,” or that this “indexing” necessarily results in an “index.” Notably, Dickinson does not disclose that the logical memory structure is or includes an index.

But even assuming the logical memory structure includes an index, claim 1

requires a “directory,” not an “index.” Petitioner simply asserts that “Dickinson’s index is a directory” (Pet., 72), but provides no evidence to support the point. Dr. Bhattacharjee, for his part, only repeats Petitioner’s conclusion, without further explanation or analysis. *See* Ex. 1003 [Bhattacharjee] ¶231. Such conclusory assertions are insufficient to establish obviousness. *See* Section III.B (citing *Magnum Oil*, 829 F.3d at 1380; 37 C.F.R. § 42.65; *Xerox*, 15).

2. Dickinson Does Not Disclose A Directory “Mapped To The Plurality Of Physical Storage Devices.”

Even if we assumed, counter-factually, that Dickinson discloses “a directory,” Petitioner fails to show that the alleged directory is “mapped to the plurality of physical storage devices.”

Petitioner contends that “Dickinson’s index is a directory *mapped to the data storage facilities D1-D4* so that for each piece of sensitive data S in the index there is a mapping to the particular locations among the data storage facilities D1-D4 where the data is stored.” Pet., 72-73 (citing Ex. 1005 [Dickinson] 16:28-31; Ex. 1003, ¶231). Dickinson, however, says nothing of the sort, and the only passage on which the Petitioner relies, Dickinson at 16:28-31, does not disclose *any* mapping between the alleged “index” and D1-D4. Paragraph 231 of Dr. Bhattacharjee’s declaration just parrots the Petition’s unsupported assertion.

Dickinson does not suggest that depository 210 is “mapped to” storage

facilities D1-D4 because Dickinson discloses that each of those data storage facilities would have its own “logical memory structure” and thus would have its own alleged index/directory, as discussed below. Thus, contrary to Petitioner’s argument, Dickinson does not disclose a depository with a single directory mapped to the plurality of storage devices (D1-D4). Per Dr. Rubin:

Petitioner asserts that Dickinson discloses a “directory” in depository 210 which is “mapped to the plurality of physical storage devices,” D1-D4. To support this proposition, Petitioner points to Dickinson’s statement that “the depository 210 comprises a single logical memory structure...” (Ex. 1005, 16:29-31) and an additional teaching that “the depository 210 may comprise distinct and physically separated data storage facilities, as disclosed further with reference to Figure 7” (*id.*, 17:4-5). Pet., 71 (“Dickinson’s depository 210, which may comprise FIG. 7’s storage facilities D1-D4, comprises a single logical memory structure.”). Thus, Petitioner seems to suggest that because depository 210 includes a single logical memory structure and may further include D1-D4 in the Figure 7 embodiment, the alleged index in depository 210 would be “mapped” to D1-D4.

But Dickinson’s discussion of Figure 7 does not suggest that each of D1-D4 is a part of a single depository 210. Rather, Figure 7 identifies **each** of the depositories in depository system 700 as its own depository 210.

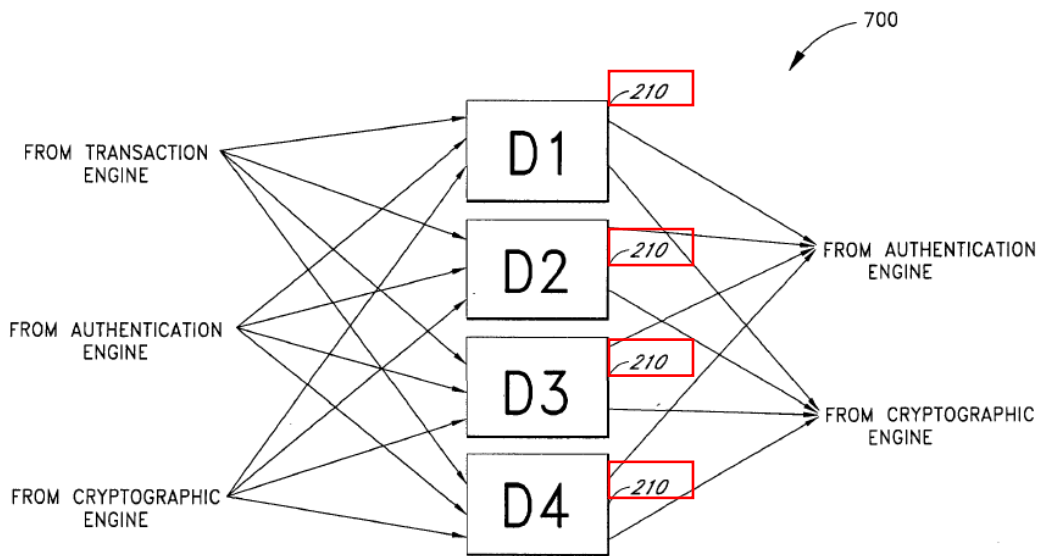


FIG. 7

Ex. 1005 [Dickinson] Fig. 7 (annotated). In the same vein, Dickinson teaches that “each of the data storage facilities D1 through D4 may advantageously comprise some or all of the elements disclosed with reference to the depository 210 of FIGURE 4.” *Id.*, 19:17-20. Dickinson further teaches that “each data storage facility, D1 through D4, comprises a separate and **independent** storage system” including in the depository system 700 embodiment. *Id.*, 20:1-3.

Consequently, each of D1-D4 is separate and independent, and would include its own depository 210 with its own “logical memory structure indexing authentication data and cryptographic key data to a unique user ID.” *Id.*, 16:30-31. Thus, there would be no need for an index or directory mapped to each of D1-D4. In accord, while each data storage facility may receive requests from various engines (the transaction engine, the authentication engine, and the cryptographic

engine), there is no teaching that any given storage facility may make requests to other storage facilities or that each instantiation of depository 210 holds a “directory” mapped to the other storage facilities. *Id.*, 19:17-24.

Ex. 2003 [Rubin-Decl.] ¶¶129-131.

Recognizing the lack of an actual disclosure, Petitioner contends that “[t]o the extent Dickinson is read to not expressly disclose a directory with this mapping,” it “would have been the conventional and obvious way to implement Dickinson’s logical memory structure so that each piece of sensitive data S included in that logical memory structure can be retrieved from data storage facilities D1-D4 when desired.” Pet., 73 (citing Ex. 1003 [Bhattacharjee] ¶231). But once again, Petitioner’s argument is simply a conclusion that relies only on Dr. Bhattacharjee’s unsupported declaration. *Compare* Pet., 72-73 with Ex. 1003 [Bhattacharjee] ¶231.

Nor is Petitioner’s contention convincing on its face. As Dr. Rubin explains:

Petitioner contends that it would be conventional and obvious to “implement Dickinson’s logical memory structure so that each piece of sensitive data S included in that logical memory structure can be retrieved from data storage facilities D1-D4 when desired.” Pet., 73. But this contention does not follow, and it is contrary to Dickinson’s disclosure. Petitioner seems to imagine a logical memory structure interacting with and retrieving data from various data storage

facilities D1-D4. But Dickinson, instead, teaches that each of storage facilities D1-D4 is a separate and independent instantiation of depository 210, each with its own logical memory structure. *See* Ex. 1005 [Dickinson] 19:16-20; 16:29-31; Fig. 7. Each of D1-D4 may receive requests directly from various engines. *Id.*, 19:20-26. The embodiments Petitioner relies upon in its discussion regarding limitation 1[a] requiring a “plurality of physical storage devices” also twice states that the facilities are “separate” and “independent.” *Id.*, 20:1-3; Pet., 68. Because each of D1-D4 operates independently, includes its own logical memory structure, and receives requests independently, there is no need for a “directory” mapped to D1-D4.

In this regard, Dickinson’s disclosure is unlike the ’140’s discussion of the “PFS directory” which “may maintain a directory that contains information about the location of the shares (for example, the file server location) and the names of the files at the share locations that include the share data.” Ex. 1001 [’140] 87:20-23.

Ex. 2003 [Rubin-Decl.] ¶¶133-134.

3. The “Physical Locations Of The Shares Are Hidden From The Client Device.”

Even if, *arguendo*, Dickinson discloses “a directory mapped to the plurality of physical storage devices” (it does not), Petitioner fails to show that the mapping occurs “such that the physical locations of the shares are hidden from the client device.”

The entirety of Petitioner’s argument is contained in a fragment of a sentence merely asserting that “the actual physical data storage facilities (*e.g.*, D1-D4) on which the data is stored are hidden from the user”:

That logical memory structure is a virtual disk under SFI’s construction because it is a virtual (‘logical’) representation of storage that appears to the user to be a physical disk but where the actual physical data storage facilities (*e.g.*, D1-D4) on which the data is stored are hidden from the user. Ex. 1005 [Dickinson] 16:28-31, 29:12-30:2; Ex. 1003 [Bhattacharjee] ¶229.

Pet., 71-72.

Petitioner, thus, says that the “actual physical data storage facilities (*e.g.*, D1-D4) on which the data is stored are hidden from the user,” but, does not prove it. Petitioner cites Dickinson at 16:28-31 and 29:12-30:2 but those passages are irrelevant. Dickinson at 16:28-31, does not mention hiding the location of shares from the client device or storage facilities D1-D4. Dickinson at 29:12-30:2 concerns a “sensitive data vault” in the context of “authentication process 1000” and its various embodiments. The only mention of data storage facilities D1-D4 in this passage is that the sensitive data vault may be “split and stored in the depository system 700.” There is no suggestion therein that the physical locations of the shares are “hidden from the client device.”

The only other support to which Petitioner cites is testimony by its expert, who simply repeats the Petition's verbiage without further elaboration. *Compare* Pet., 72 *with* Ex. 1003 [Bhattacharjee] ¶229. Such conclusory testimony deserves little or no weight. *See* Section III.B (37 C.F.R. § 42.65; *Xerox*, 15).

Dickinson's lack of disclosure, moreover, lies in stark contrast with the '140's clarity, providing that "[t]he underlying distribution of shares ***may be hidden from users/clients***" and "[t]he PFS provides an interface that allows users to access parsed files ***without knowledge of the location of shares***." Ex. 1001 ['140] 87:35-39.

Thus, Petitioner's conclusory assertion that Dickinson hides the location of shares from the client device cannot establish obviousness.

4. Dickinson Does Not Disclose "A Secure Storage System Configured To[] Present To A Client Device A Virtual Disk."

Limitation 1[b][1] additionally requires "a ***secure storage system configured to: present to a client device a virtual disk***" Petitioner identifies Dickinson's data vault and transaction engines as comprising the "secure storage system." Pet., 70-71. Petitioner further identifies Dickinson's "logical memory structure" as the "virtual disk." Pet., 71-72 ("[Dickinson's] logical memory structure is a virtual disk ..."). Petitioner, thus, must show that Dickinson's data vault engine and transaction engine comprise a "secure storage system" that "present[s]" the

“logical memory structure” to the client device. Petitioner’s argument fails, however, as it fails to show that Dickinson’s logical memory structure is “presented” to the client device much less that it is presented by the “secure storage system.”

As Dr. Rubin explains, Petitioner does not prove that the logical memory structure is “present[ed] to the client device”:

Petitioner contends the “logical memory structure” is presented to the client device by the “transaction engine” “[w]hen the user system 105 requests the sensitive data from the sensitive data vault.” Pet., 73. Petitioner, however, does not prove this.

There is, quite simply, no reason to believe that the **logical memory structure** is presented to the user when the user requests sensitive data. When user system 105 requests data from the sensitive data vault of the “data vault engine,” Dickinson teaches that an authentication process commences. Ex. 1005 [Dickinson] 29:17-19. Therein, the “authentication engine” compares the user’s login information with its enrollment information in depository 210. *Id.*, 15:4-12. If the user is authenticated, **sensitive data from the sensitive data vault is served to the user**. *Id.*, 29:25-30.

Thus, when the user requests sensitive data, upon verification, “sensitive data ... is served to the user.” There is no indication, however, that **the “logical memory structure”** (Petitioner’s alleged “virtual disk”) is presented to the user. The sensitive data and the

logical memory structure are two different things. The logical memory structure stores “[enrollment] authentication data and cryptographic key data [for] a unique user ID.” Ex. 1005 [Dickinson] 16:28-31; *id.*, 14:17-19. Its role is to facilitate the authentication of users seeking to access sensitive data. It, itself, is highly sensitive as it contains login/authentication information for a specific user. Dickinson does not disclose “presenting” the logical memory structure to the user and it’s not self-evident why it would.

Ex. 2003 [Rubin-Decl.] ¶¶142-144.

Petitioner further contends that it is the transaction engine that “presents” the logical memory device to the client. *See* Pet., 73 (“...the logical memory structure (‘virtual disk’) is presented to the [client device] **by the transaction engine...**”).

This is incorrect, as Dr. Rubin explains:

Contrary to Petitioner’s contention (Pet., 73), Dickinson does not disclose that the transaction engine “presents” the “logical memory structure” (in depository 210) to the client device. Dickinson discloses that the transaction engine “comprises a data routing device,” that may route data **to** “one of the depository 210, the authentication engine 215, the cryptographic engine 220, and the mass storage 225.” Ex. 1005 [Dickinson] 13:18-19, 13:23-25. The transaction engine also routes data from the authentication and cryptographic engines to the user system 105. *Id.*, 13:25-28.

Dickinson, however, makes no mention of routing the logical memory structure, or anything else, **from depository 210** to the client

device. As Figure 2 further illustrates, the transaction engine 205 may route requests to depository 210, *but*, there is no indication the transaction engine presents depository 210 or its logical memory structure to the client device:

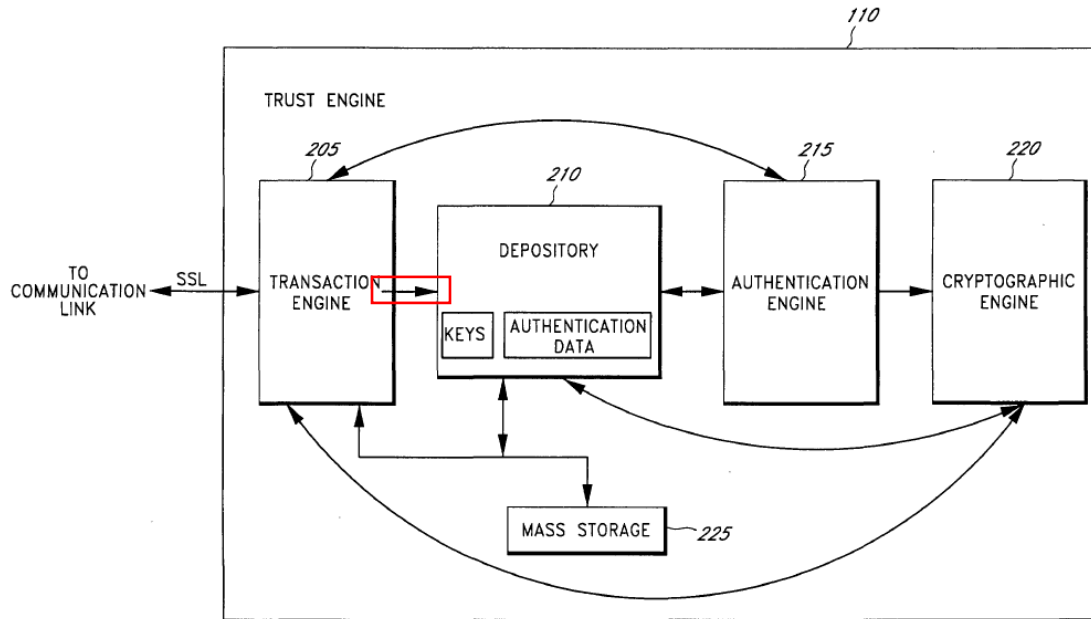


FIG. 2

Ex. 1005 [Dickinson] Fig. 2 (annotated). The one-way arrow between the transaction engine and depository 210 indicates that the transaction engine may transmit data to depository 210 but nothing identified in this figure or Dickinson's disclosure suggests the logical memory structure or anything else in depository 210 is presented by the transaction engine to the client device/user.

In accord, Figure 4 further confirms that depository 210 receives data from the transaction engine but does not send data to the transaction engine:

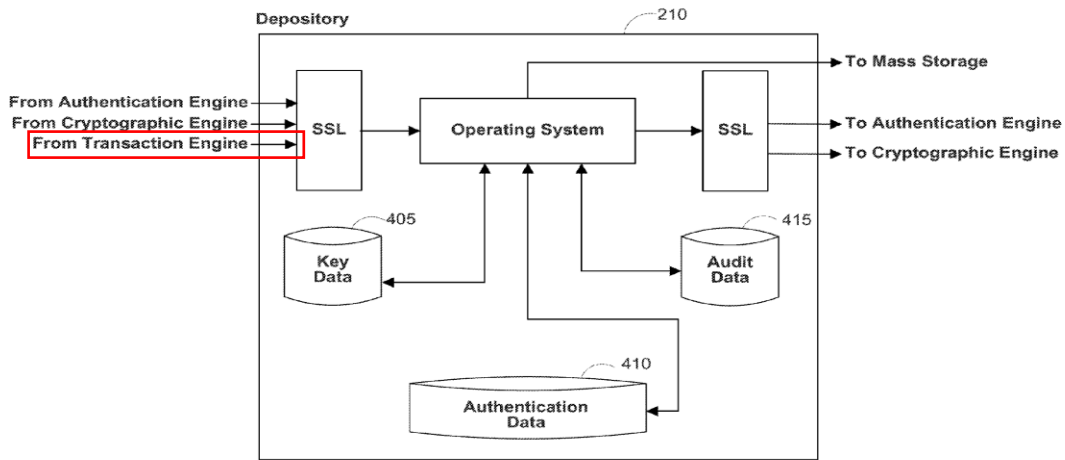


FIG. 4

Ex. 1001 [’140] Fig. 4.

Ex. 2003 [Rubin-Decl.] ¶¶146-148.

Petitioner further contends that “a POSA would have understood that in the embodiment that has a data vault engine, the data stored therein is included in the index of the user’s data.” Pet., 72. This contention is ultimately irrelevant, as even if the data stored in the data vault engine was included in the purported “index,” it does not show the presentation of the logical memory structure to the client device by a “secure storage system.”

Petitioner alternatively contends that even if Dickinson does not “expressly disclose that the logical memory structure is presented to the user system 105, that would have been an obvious way to implement what Dickinson describes ...” Pet., 72-74. Petitioner’s contention, however, is ultimately irrelevant, because the

claims require a “**secure storage system** configured to” “present to a client device a virtual disk.” Thus, even if, *arguendo*, a POSITA were motivated, for some reason, to present the logical memory structure to the client device, there is no teaching that this would occur vis-à-vis the transaction engine and/or the data vault engine, which are the only components Petitioner contends comprise the secure storage system.

Moreover, there is no indication that a POSITA would be so motivated. As Dr. Rubin explains:

Under Dickinson’s current structure, as described above, an authentication engine compares a login attempt with enrollment data to determine whether a user login attempt is successful. Ex. 1005 [Dickinson] 17:6-35. There is no reason to alter Dickinson such that the user device would have direct access to the logical memory structure which contains both “authentication data and cryptographic key data.” *Id.*, 16:28-31. This would result in a system that is less secure and for no discernible benefit.

Ex. 2003 [Rubin-Decl.] ¶154.

Petitioner further argues that including “the data stored” in the data vault engine “in the index of the user’s data ... would have been the obvious way to implement what Dickinson describes so that the user can see all the user’s stored data in the index and retrieve the data the user desires to access.” Pet., 72. This

argument is similarly irrelevant, as Petitioner still fails to even contend it would be obvious for the “secure storage system” (*e.g.*, the transaction engine) to present a logical memory structure “indexing authentication data and cryptographic key data to a unique user ID” (Ex. 1005, 16:28-31) to the client device. And, as just explained, to the extent the “logical memory structure” were presented to the client device, it would needlessly result in a system that is less secure.

D. Dickinson-Takeda Does Not Disclose Or Render Obvious Limitation 1[b][3].

Limitation 1[b][3] requires a “plurality of shares” “associated with at least one session key used to secure a data set” (limitation 1[a]) wherein “data indicative of the at least one session key” is “include[d] with each of the plurality of shares.” The teachings relating to “session keys” in general, and limitation 1[b][3] in particular, are a central focus of the material added to the ’140 over Dickinson. *See* Section IV.A.2(b), *supra*. Petitioner contends that limitation 1[b][3] is disclosed by Dickinson alone (contrary to the Examiner’s determination, *see* Section IV.B, *supra*) or by Dickinson in combination with Takeda. Petitioner is wrong on both counts.

1. Dickinson Alone Does Not Disclose Or Render Obvious Limitation 1[b][3].

Petitioner argues that Dickinson, alone and without modification, discloses limitation 1[b][3]. Pet., 60, 64, 74. Petitioner relies on Dickinson’s Figure 8 and

alleges that random numbers “A” and “C” are session keys. *See, e.g.,* Pet., 68-70, 75-76. But all of Dickinson’s discussion of “session keys” arises in relation with Figure 12, which Petitioner does not rely upon. *See* Section IV.A.2(b).

Even assuming that A and C are session keys, Dickinson does not disclose including data indicative of the session keys with the shares. As Dr. Rubin explains:

The Applicant in prosecution pointed out that Dickinson describes a data splitting process 800 in which *transformed versions* of an *entire* set of data ‘S,’ not data indicative of session keys used to secure the dataset, are stored in multiple data storage facilities. *See* Ex. 1002 [Pros.-Hist.] 1245-6 (citing Ex. 1005 [Dickinson] 21-22, Fig. 8). Based on Dickinson’s described process, this is accurate. Dickinson discloses in Figure 8 that dataset “S” is received and a data splitting module generates random numbers “A” and “C.” Ex. 1005 [Dickinson] 20:23-32. “S” is “transformed” because it is combined in whole with the number “A” and again with the number “C” resulting in the new numbers “B” and “D.” Dickinson explains that the “number B may comprise the binary combination of A XOR S and [the] number D may comprise the binary combination of C XOR S.” Ex. 1005 [Dickinson] 20:33-34. No data from the purported “session keys,” “A” and “C,” are included within the purported shares, “B” and “D.” Instead, “A” and “C” are separate files that must be separately stored and recovered to reassemble dataset “S.” *See id.*, 21:1-14.

Thus, both “B” and “D” are each “transformed” versions of the entire dataset “S”; they do not teach or suggest storing ***any additional information*** with each of the data shares, much less information used to secure a dataset (*i.e.*, “data indicative of a session key”).

Ex. 2003 [Rubin-Decl.] ¶¶160-161.

2. The Combination Of Dickinson And Takeda Does Not Disclose Or Render Obvious Limitation 1[b][3].

Petitioner also contends that limitation 1[b][3] is disclosed by combining Dickinson and Takeda in two ways that it identifies as “alternatives (2) and (3).” Pet., 59-60. These alternatives fail because Petitioner does not, and cannot, show a motivation to combine Dickinson and Takeda. Petitioner contends a POSITA would be motivated to modify Dickinson by replacing Dickinson’s “A” and “C” (the alleged session keys) with Takeda’s pointers to and/or metadata relating to those alleged keys. Pet., 60, 64. Petitioner contends a POSITA would be motivated to modify Dickinson because “Dickinson does not specify how its system knows where to retrieve the correct key ... to recover S,” that “storing, with encrypted data, a pointer to and/or metadata about a key ... beneficially allows for the separate management of the encrypted data and the keys for that data, for example to best use limited storage space,” and that storing the pointer to and/or metadata of a key with its respective share “provides a technique for providing the

association between the encrypted data portion and its key” where “[t]he stored metadata could also be used beneficially during decryption.” Pet., 61.

Petitioner’s proffered motivations should be rejected because (1) they do not provide any benefits over Dickinson’s approach, and (2) their implementation would negatively alter Dickinson’s design, which specifically contemplates “redundancy” and “randomly distribut[ing]” the initial four pairings “to one of the depositories D1 through D4.” Ex. 1005 [Dickinson] 20:4-7, 20:9-11.

“The Board must weigh the benefits and drawbacks of the modification against each other, to determine whether there would be a motivation to combine.” *Arctic Cat*, 795 Fed. App’x. at 833; *Henny Penny*, 938 F.3d at 1331-32 (similar). As Dr. Rubin explains, each of Petitioner’s proffered motivations offers no benefit and entails distinct drawbacks:

Petitioner and Dr. Bhattacharjee contend that a POSITA would be motivated to modify Dickinson’s storage of data shares and session keys in “depository system 700” such that “a pointer to and/or metadata of” a session key is stored with a data share instead of a session key itself. Pet., 61. Petitioner bases this argument on the contention that Dickinson “does not specify how its system knows where to retrieve the correct key ... for decrypting the corresponding encrypted data portion ... to recover S.” Pet., 61 (citing Ex. 1003 [Bhattacharjee] ¶204). However, Dickinson is intentionally designed such that “only data from two of the multiple data storage facilities,

D1 through D4, are needed to decipher and reassemble the sensitive data,” dataset “S.” Ex. 1005 [Dickinson] 20:4-7. This configuration “provides redundancy along with additional security measures,” while also allowing the “data stored in each data storage facility [to be] randomized and undecipherable, [so that] compromise of any individual data storage facility does not necessarily compromise the sensitive data.” *Id.*, 20:4-7, 20:9-11. This design obviates the need for Dickinson’s system to “know” where to retrieve the session keys, as redundancy and randomization are intentionally contemplated.

Petitioner and Dr. Bhattacharjee further contend a POSITA would be motivated to modify Dickinson with Takeda to “beneficially allow[] for separate management of the encrypted data and the keys for that data, for example to best use limited storage space.” Pet., 61. But the storage of a pointer to and/or metadata of a key, rather than the key itself, would come at the cost of the specifically contemplated benefit of redundancy addressed earlier. Neither Petitioner nor Dr. Bhattacharjee suggest *where* the actual keys would be stored in the proposed combination, aside from a citation to a different reference, Ophir, indicating it may be placed in some “external storage.” Pet., 61 (citing Ex. 1007 [Ophir] ¶32). But this storage scheme would result in the loss of Dickinson’s ability to reassemble the dataset by only recovering data from two of the four data storage facilities. It may even further complicate Dickinson’s recovery process, as Dickinson would need to access an undisclosed storage facility outside of its current storage scheme, which may be less secure than its depositories, to recover random numbers “A” and “C.” This

drawback would dissuade a POSITA from modifying Dickinson as Petitioner and its expert contemplate.

Petitioner and its expert also assert that a POSITA would be motivated to combine references because the metadata could be used “beneficially during decryption.” Pet., 61. But this is no motivation, as Dickinson’s random numbers “A” and “C” themselves, as the supposed “keys,” would be at least as beneficial to decrypt the “shares” as mere metadata. Petitioner and Dr. Bhattacharjee fail to explain exactly *why* the metadata for a key would be preferable to the key itself in terms of decryption. See *id.*; Ex. 1003 [Bhattacharjee] ¶207.

Petitioner and its expert also fail to disclose any benefit associated with the provision of “a technique for providing the association between the encrypted data portion and its key” such that “the correspondence relation between the data can be recognized” would render. Pet., 61. Petitioner simply asserts, without explanation, that the combination “would have motivated POSAs to use that technique to provide that association in Dickinson.” Pet., 62 (citing to Ex. 1003 [Bhattacharjee] ¶209). Petitioner’s expert also opines at ¶209, without explanation, that “a POSA...would need a way to associate data with corresponding keys in order to reconstruct the original data.” But, as discussed above, Dickinson as configured can already reconstruct the original data by recovering from only two of four storage facilities. Ex. 1005 [Dickinson] 20:6-7.

Ex. 2003 [Rubin-Decl.] ¶¶164-167.

In sum, each of Petitioner’s arguments centered upon Dickinson fail. The Examiner correctly found that Dickinson does not satisfy two distinct limitations present in each of the claims and Petitioner does not attempt to demonstrate any “material error” on the Examiner’s part. Should these shopworn arguments be revisited, Petitioner again fails to show that Dickinson satisfies limitation 1[b][1] for four independently sufficient reasons. *See* Section IV.C. Each of Petitioner’s arguments for limitation 1[b][3] similarly fail, and for the same reasons accepted by the Examiner. Nor can Petitioner’s strained combination with Takeda salvage its arguments because a POSITA would not be motivated to make the combination.

V. CONCLUSION

Petitioner fails to demonstrate a reasonable likelihood of success as to any of the challenged claims, and the Petition should be denied.

Respectfully submitted,

/ Stephen J. Elliot /

Stephen J. Elliott (Reg. No. 52,858)
SULLIVAN & CROMWELL LLP

Kenneth J. Weatherwax (Reg. No. 54,528)
Nathan Lowenstein, *pro hac vice* to be filed
Colette Woo, *pro hac vice* to be filed
LOWENSTEIN & WEATHERWAX LLP
Date: March 28, 2024

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITS

This Patent Owner Preliminary Response (the “POPR”) consists of 13,788 words, excluding table of contents, table of authorities, certificate of service, this certificate, or table of exhibits. The POPR complies with the type-volume limitation of 14,000 words as mandated in 37 C.F.R. § 42.24. In preparing this certificate, counsel has relied on the word count of the word-processing system used to prepare the paper (Microsoft Word).

Respectfully submitted,

/ Robert Pistone /

Date: March 28, 2024

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the following documents were served
by electronic service, by agreement between the parties, on the date below:

PATENT OWNER PRELIMINARY RESPONSE

EXHIBITS 2003, 2005-2006

The names and addresses of the parties being served are as follows:

Elisabeth H. Hunt	EHunt-PTAB@wolfgreenfield.com
Richard F. Giunta	RGiunta-PTAB@wolfgreenfield.com
Thomas A. Franklin	TFranklin-PTAB@wolfgreenfield.com

Respectfully submitted,

/ Colette Woo /

Date: March 28, 2024