

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner,

v.

SECURITY FIRST INNOVATIONS, LLC,
Patent Owner.

Case IPR2024-00214
Patent 11,068,609

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. THE ORSINI GROUND FAILS (ALL CLAIMS, GROUND 1).....	1
A. The '609 Patent Claims Fundamentally Different Technology Than That Disclosed In Orsini.	1
B. Petitioner Fails To Show That Orsini Anticipates “Performing A Cryptographic Operation Based On The First Key To Further Secure The Plurality Of Data Chunks” (Limitation 1[e]).	7
1. Orsini Does Not Expressly Disclose Claim 1.....	8
2. Orsini Does Not Inherently Disclose Claim 1.	13
C. Petitioner Fails To Show That Orsini Renders The Claims Obvious.	19
1. Petitioner Cannot Rely On Common Sense To Supply Missing Limitation 1[e].	20
2. Petitioner Fails To Show That A POSITA Would Have Been Motivated To Modify Orsini to Add Limitation 1[e].....	22
III. THE FOSTER GROUND FAILS (ALL CLAIMS, GROUND 2).....	36
A. The '609 And Foster Fundamentally Differ.....	36
B. Petitioner Fails To Show Foster Discloses “Generating A Plurality Of Data Chunks Based On A Data Set” (Limitation 1[c][1]).	39
C. Petitioner Fails To Show That Foster Renders Obvious “The Data Set Can Be Reconstructed Using At Least A Minimum Number Of The Plurality Of Chunks” (Limitation 1[c][2]).	45

D. Petitioner Fails To Show That Foster Discloses Or Renders
Obvious “Storing, . . . At Least One Data Chunk . . . On At Least
One Storage Device” (Limitation 1[f]). 54

E. Petitioner Fails To Show That Foster Teaches The Claimed Steps
In Order. 62

IV. CONCLUSION.....66

TABLE OF AUTHORITIES

Page(s)

COURT DECISIONS

<i>Archer Daniels Midland Co. v. United States</i> , 561 F.3d 1308 (Fed. Cir. 2009)	58
<i>Arctic Cat Inc. v. Bombardier Rec. Prods.</i> , 876 F.3d 1350 (Fed. Cir. 2017)	27
<i>Arctic Cat Inc. v. Polaris Indus.</i> , 795 Fed. App'x. 827 (Fed. Cir. 2019)	27, 62
<i>Arendi S.A.R.L. v. Apple, Inc.</i> , 832 F.3d 1355 (Fed. Cir. 2016)	20
<i>Ariosa Diagnostics v. Verinata Health, Inc.</i> , 805 F.3d 1359 (Fed. Cir. 2015)	32
<i>Ashland Oil, Inc. v. Delta Resins & Refractories, Inc.</i> , 776 F.2d 281 (Fed. Cir. 1985)	64
<i>ATD Corp. v. Lydall, Inc.</i> , 159 F.3d 534 (Fed. Cir. 1998)	52
<i>BlephEx, LLC v. Myco Indus.</i> , 24 F.4th 1391 (Fed. Cir. 2022)	18
<i>D Three Enters., LLC v. SunModo Corp.</i> , 890 F.3d 1042 (Fed. Cir. 2018)	13
<i>DSS Tech. Mgm't v. Apple Inc.</i> , 885 F.3d 1367 (Fed. Cir. 2018)	20
<i>Enplas Display Device Corp. v. Seoul Semiconductor Co., Ltd.</i> , 909 F.3d 398 (Fed. Cir. 2018)	13

<i>Finisar Corp. v. DirecTV Group, Inc.</i> , 523 F.3d 1323 (Fed. Cir. 2008)	19
<i>Forum US, Inc. v. Flow Valve, Ltd. Liab. Co.</i> , 926 F.3d 1346 (Fed. Cir. 2019)	13
<i>Henny Penny Corp. v. Frymaster LLC</i> , 938 F.3d 1324 (Fed. Cir. 2019)	27, 62
<i>HTC Corp. v. Cellular Commns. Equip., LLC</i> , 877 F.3d 1361 (Fed. Cir. 2017)	15
<i>IBM v. Iancu</i> , 759 Fed. App'x. 1002 (Fed. Cir. 2019)	58
<i>In re Benno</i> , 768 F.2d 1340 (Fed. Cir. 1985)	13
<i>In re Conrad</i> , 759 Fed. App'x. 982 (Fed. Cir. 2019)	35
<i>In re Kahn</i> , 441 F.3d 977 (Fed. Cir. 2006)	63
<i>In re Magnum Oil Tools Int'l, Ltd.</i> , 829 F.3d 1364 (Fed. Cir. 2016)	52, 64
<i>In re NTP, Inc.</i> , 654 F.3d 1279 (Fed. Cir. 2011)	26, 52
<i>In re Ratti</i> , 270 F.2d 810 (CCPA 1959)	62
<i>Intel Corp. v. PACT XPP Schweiz AG</i> , 61 F.4th 1373 (Fed. Cir. 2023)	33, 47
<i>InTouch Techs., Inc. v. VGo Commc'ns., Inc.</i> , 751 F.3d 1327 (Fed. Cir. 2014)	35
<i>KSR Int'l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007).....	34, 63

<i>LBT IP I LLC v. Apple Inc.</i> , 2023 U.S. App. LEXIS 14376 (Fed. Cir. 2023)	15
<i>Mantech Envtl. Corp. v. Hudson Envtl. Servs.</i> , 152 F.3d 1368 (Fed. Cir. 1998)	7
<i>Microsoft Corp. v. Biscotti, Inc.</i> , 878 F.3d 1052 (Fed. Cir. 2017)	15
<i>Net MoneyIN, Inc. v. VeriSign, Inc.</i> , 545 F.3d 1359 (Fed. Cir. 2008)	18
<i>Otsuka Pharm. Co., v. Sandoz, Inc.</i> , 678 F.3d 1280 (Fed. Cir. 2012)	35
<i>Plas-Pak Indus., Inc. v. Sulzer Mixpac AG</i> , 600 F. App'x 755 (Fed. Cir. 2015)	62
<i>Polaris Indus., Inc. v. Arctic Cat, Inc.</i> , 882 F.3d 1056 (Fed. Cir. 2018)	61
<i>Ruiz v. A.B. Chance Co.</i> , 357 F.3d 1270 (Fed. Cir. 2004)	35, 52
<i>Schumer v. Lab. Computer Sys., Inc.</i> , 308 F.3d 1304 (Fed. Cir. 2002)	16
<i>Univ. of Strathclyde v. Clear-Vu Lighting LLC</i> , 17 F.4th 155 (Fed. Cir. 2021)	35

AGENCY DECISIONS

<i>Apple Inc. v. Cellular Comm'ns Equipment, LLC</i> , IPR2015-00576, Paper 7 (June 12, 2015)	26, 52
<i>Carefusion Corp. v. Baxter Int'l, Inc.</i> , IPR2016-01456, Paper 9 (Feb. 6, 2017)	53

<i>Edwards Lifesciences Corp. v. Aortic Innovations LLC</i> , IPR2023-01232, Paper 10 (Feb. 7, 2024).....	33, 47
<i>F5 Networks, Inc. v. Radware, Ltd.</i> , IPR2017-00124, Paper 48 (Apr. 23, 2018).....	27, 52
<i>Hologic, Inc. v. Enzo Life Sciences, Inc.</i> , IPR2018-00019, Paper 21 (Nov. 28, 2018)	53
<i>Parsons Xtreme Golf, LLC v. Taylor Made Golf Co., Inc.</i> , IPR2018-00518, Paper 14 (July 18, 2018)	33
<i>RPX Corp. v. Parity Networks, LLC</i> , IPR2018-00097, Paper 7 (Apr. 24, 2018).....	33, 47
<i>Samsung Elecs. Co., Ltd. v. Smart Mobile Techs. LLC</i> , IPR2022-01248, Paper 53 (Jan. 23, 2024).....	51
<i>Samsung Elecs. v. Uniloc 2017 LLC</i> , IPR2020-00046, Paper 6 (Apr. 1, 2020).....	53
<i>Toyota Motor Corp. v. Blitzsafe Texas, LLC</i> , IPR2016-00422, Paper 12 (July 6, 2016)	53
<i>Xerox Corp. v. Bytemark, Inc.</i> , IPR2022-00624, Paper 9 (Aug. 24, 2022) (precedential)	40, 64

REGULATIONS

37 C.F.R. § 42.104(b)(3).....	53
37 C.F.R. § 42.104(b)(4).....	53
37 C.F.R. § 42.6(a)(3)	16
37 C.F.R. § 42.65(a).....	64

OTHER AUTHORITIES

*Changes to the Claim Construction Standard for Interpreting Claims in Trial
Proceedings Before the Patent Trial and Appeal Board,*
83 Fed. Reg. 51,340 (Oct. 11, 2018) 54

EXHIBIT LIST	
2001	Reserved
2002	Reserved
2003	Declaration of Aviel D. Rubin, Ph.D.
2004	Prosecution History of 10,452,854 Patent ['854-FH]
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]
2007	Excerpts from Jonathan Katz et al., <i>Introduction to Modern Cryptography</i> (1st ed. 2007) [Katz]
2008	Bruce Schneier, <i>Security Pitfalls in Cryptography</i> , SCHNEIER ON SECURITY, 1998, https://www.schneier.com/essays/archives/1998/01/security_pitfalls_in.html [Schneier-Security-Pitfalls-in-Cryptography]
2009	Excerpts from Ross Anderson, <i>Security Engineering: A Guide to Building Dependable Distributed Systems</i> (1st ed. 2001) [Anderson]
2010	Excerpts from Bruce Schneier, <i>Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C</i> (2nd ed. 2015) [Schneier-Applied-Cryptography]
2011	Patent Application Publication to Tsai US 2005/0195857 A1 [Tsai]
2012	Edgar Pek et al., <i>Formal Verification of Communication Protocols in Distributed Systems</i> , SYMPOSIUM ON OPERATING SYS. PRINCIPLES, 2003 [Pek]

2013	Jan Friso Groote et al., <i>A Bounded Retransmission Protocol for Large Data Packets: A Case Study in Computer Checked Algebraic Verification</i> , 1996 [Groote]
2014	Microsoft, <i>Overview of MPEG-2 Systems</i> , WWW.LEARN.MICROSOFT.COM , July 27, 2023, https://learn.microsoft.com/en-us/windows/win32/directshow/overview-of-mpeg-2-systems [Overview-of-MPEG-2-Systems]
2015	Excerpts from Harry Newton, <i>Newton's Telecom Dictionary</i> (16th ed. 2000) [Newton's-Telecom-Dictionary]
2016	Excerpts from <i>Oxford Dictionary of Computing</i> (1996) [Oxford-Dictionary-of-Computing]
2017	IBM, <i>Transmission Control Protocol (TCP)</i> , WWW.IBM.COM , https://www.ibm.com/docs/en/zos-basic-skills?topic=4-transmission-control-protocol-tcp [IBM]
2018	<i>Security First Innovations, LLC v. Google LLC</i> , 2:23-cv-00091 (JKW-RJK), Google's Opening Claim Construction Brief (December 6, 2023) [Google's-Opening-Claim-Construction-Brief]
2019	Excerpts from Microsoft Press, <i>Microsoft Computer Dictionary</i> (5th ed. 2002) [Microsoft-Computer-Dictionary]
2020	<i>Security First Innovations, LLC v. Google LLC</i> , 2:23-cv-00091 (JKW-RJK), Google's Responsive Claim Construction Brief (Non-Confidential Redacted Version) (Jan. 8, 2024) [Google-Responsive-Claim-Construction-Brief]
2021	Cambridge Dictionary, <i>Set-Top Box</i> , https://dictionary.cambridge.org/us/dictionary/english/set-top-box (last visited Feb. 27, 2024)

2022	Jim Connors, INFOWORLD, <i>The evolution of cable television to interactive communications service provider</i> (Oct. 1, 1996), https://www.infoworld.com/article/2077255/the-evolution-of-cable-television-to-interactive-communications-service-provider.html [InfoWorld]
------	---

I. INTRODUCTION

The Petition should be denied as it fails to demonstrate a reasonable likelihood of prevailing as to at least one of the challenged claims under either of the grounds on which Petitioner relies. *See* Sections II-III.

II. THE ORSINI GROUND FAILS (ALL CLAIMS, GROUND 1).

Petitioner's first ground argues that the claims are anticipated and/or rendered obvious by Orsini. Pet., 2. Orsini was disclosed to and considered by the Examiner. Ex. 2004 ['854-FH] 45. The Examiner allowed the claims over Orsini because the claimed inventions are not disclosed in Orsini. *See* Section II.A. Petitioner's anticipation (Section II.B) and obviousness (Section II.C) arguments fail.

A. The '609 Patent Claims Fundamentally Different Technology Than That Disclosed In Orsini.

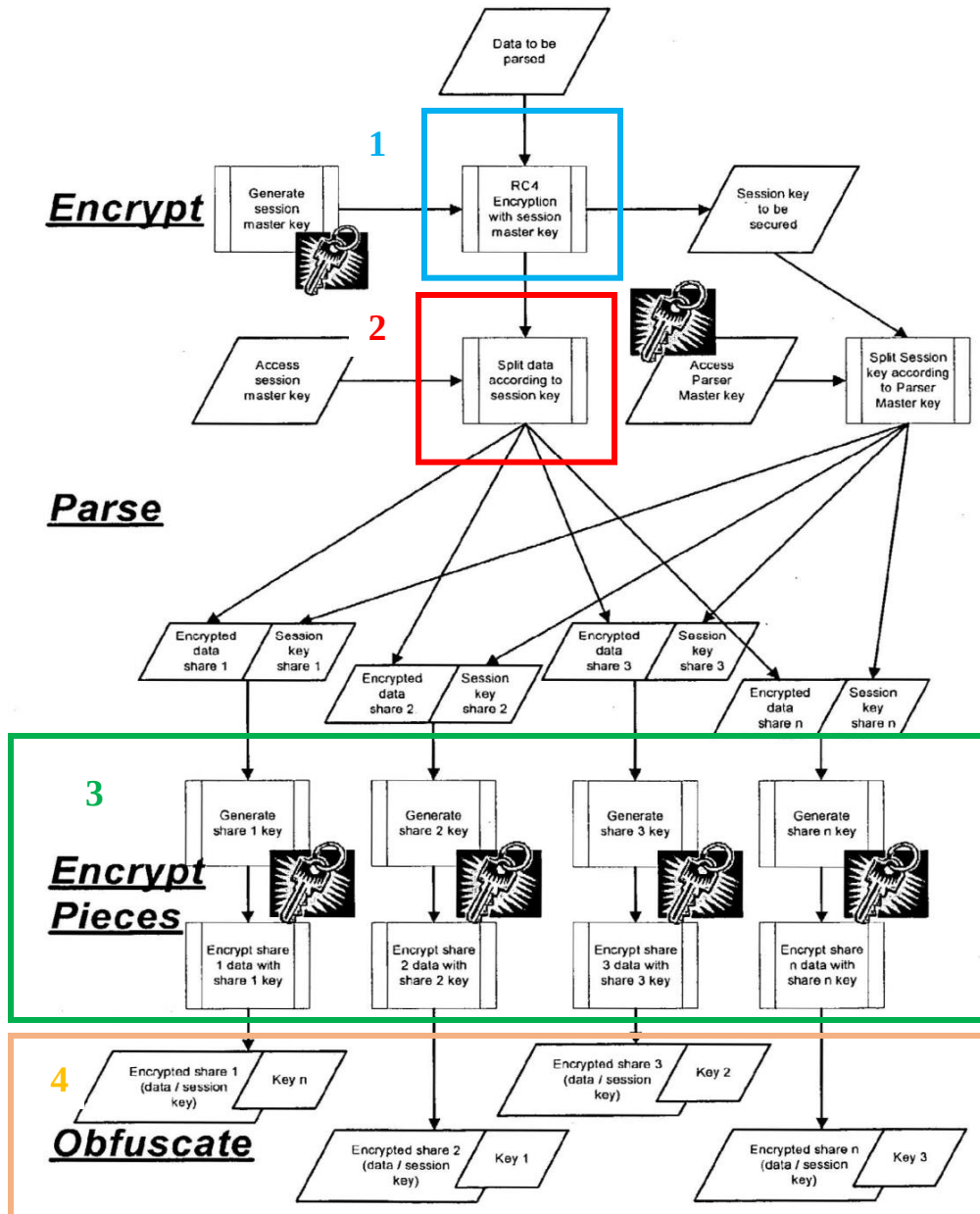
Orsini is Patent Owner's application and shares three (of four) named inventors. Figures 1-25 and paragraphs 1-423 of Orsini are included in the '609's specification. The '609, however, includes ***17 new figures and over eighteen columns of additional material*** (~10,500 words) which contain significantly different technology than that disclosed in Orsini. *See* Ex. 1001 ['609] Figs. 26-41, 6:44-6:62, 8:10-8:61, 65:56-83:3.

Petitioner's Ground 1 relies, almost entirely, on the dataset-securing method disclosed in Orsini's Figure 21. *E.g.*, Pet., 9-16, 18, 27-28, 29, 43. Orsini's Figure 21 and the '609's claim 1 are each directed to methods for securing a "dataset." Both methods rely, in part, on dividing the dataset into pieces (called "chunks" or "shares" in the '609) and encrypting the chunks.

From here, however, the methods diverge. As Dr. Aviel Rubin explains, Figure 21's method of securing data is centered around encrypting a dataset with a "session master key," parsing the dataset and the session master key into shares, applying a share key to each of the shares, and then non-cryptographically storing the share keys with a share it was not used to encrypt:

As can be seen below, Figure 21 teaches **(1) encrypting the entire data set with the session master key**, **(2) parsing the data set and session master key into shares**, then **(3) encrypting those shares and split session master key with a set of share keys**, followed by a **(4) non-cryptographic** step:

Figure 21



Ex. 1005 [Orsini] Fig. 21 (annotated), [0293].

...

The end result of Orsini's Figure 21's method is that a piece of the "session master key" used to encrypt the original, undivided dataset

is stored in each of the shares, and all of the shares (including a piece of the session master key) are themselves encrypted with a “share key.” Each share key is then hidden by storing it with a share that the key cannot decrypt. Thus, to reconstruct a dataset secured by Figure 21’s method, one can: (a) collect each of the shares together, (b) use the unencrypted share keys to decrypt the pieces of the encrypted dataset and session master key in the shares, (c) reconstruct the encrypted data set and the session master key, and (d) use the reconstructed session master key to decrypt the dataset.

Ex. 2003 [Rubin-Decl.] ¶¶ 35-37.

The ’609’s claim 1, by contrast, is rooted in the inventors’ insight that Orsini’s method potentially allows an attacker to reconstruct the dataset if the attacker has access to each of the shares; the ’609’s claimed inventions eliminate that potential vulnerability. As Dr. Rubin explains:

Orsini’s Figure 21 merely hides the keys used to encrypt the data chunks by storing each key with a chunk it was not used to encrypt. Ex. 1005 [Orsini] [0299]. Orsini’s non-cryptographic approach, however, cannot protect a dataset if an attacker has access to all of the shares (and thus access to all of the share keys) and can therefore reveal the share keys and use them to decrypt the corresponding shares. In that case, reconstruction of the original dataset becomes straight-forward. Orsini’s Figure 21 does not teach any method of “further securing” the dataset after the data shares have been encrypted with the share keys other than by hiding the share keys with different shares.

Ex. 2003 [Rubin-Decl.] ¶ 39.

In contrast, the '609's claim 1 concerns a method, wherein a workgroup key (the claimed "first key") further **cryptographically** secures the shares/chunks preventing an attacker from reconstructing the dataset even with access to all of the shares. This innovation is embodied in the limitation of claim 1 requiring "performing a cryptographic operation based on the first key to further secure the data set." As Dr. Rubin explains:

The '609 teaches further securing the share keys cryptographically by encrypting them "with a workgroup key," explaining that "[t]he workgroup key concept allows for enhanced protection of information assets by encrypting information stored within the shares." Ex. 1001 ['609] 79:60-62. Beneficially, "[o]nce this operation is performed, even if all required shares and external keys are discovered, an attacker has no hope of recreating the information without access to the workgroup key." *Id.*, 79:62-65. Thus, the '609 introduces a "first key" (*i.e.*, the workgroup key) used in a cryptographic operation to further secure data, and teaches the benefits of this approach. *See also id.*, Fig. 39, 80:24-27.

Ex. 2003 [Rubin-Decl.] ¶ 40.

In short, the '609 discloses and claims a materially different and improved method for securing a dataset from that disclosed in Orsini that addresses a potential weakness in Figure 21's method. Claim 1, *inter alia*, uses key-based

cryptography to further secure the encrypted shares after they are created, a step that is not disclosed in Orsini (which merely hides the keys used to encrypt the shares among the shares themselves). Thus, Orsini does not disclose the new step of “performing a cryptographic operation based on the first key” after the data set has been split into shares/chunks and those shares have been individually encrypted with a second key. Ex. 2003 [Rubin-Decl.] ¶ 41.

Moreover, according to Petitioner’s mapping, Figure 21’s method is simply incompatible with claim 1 and cannot render it obvious. Petitioner equates the claimed “first key” (*i.e.*, the workgroup key) with the “session master key” of Orsini’s Figure 21. In other words, under Petitioner’s interpretation, the operation to “further secure” the dataset after encrypting the shares is based on the session master key. But, as discussed above, in Figure 21, the session master key is divided into pieces, the pieces are integrated into the shares, and then the session master/share combinations are encrypted with the share keys. Thus, by the time the dataset is to be “further secured” as required by claim 1, the session master key in Orsini is buried within encrypted shares, and any further use of it would entirely undermine the prior steps in Orsini’s method. Ex. 2003 [Rubin-Decl.] ¶ 42. If the session master key is used in a “cryptographic operation” to “further secure the plurality of data chunks” as Petitioner suggests, a priori knowledge of the session master key would be needed to unpack the encrypted shares. This would require

complex changes to Figure 21's method due to the need for first key management and would further remove the benefits of securely storing and transporting the session master key within the encrypted shares. *Id.*

B. Petitioner Fails To Show That Orsini Anticipates “Performing A Cryptographic Operation Based On The First Key To Further Secure The Plurality Of Data Chunks” (Limitation 1[e]).

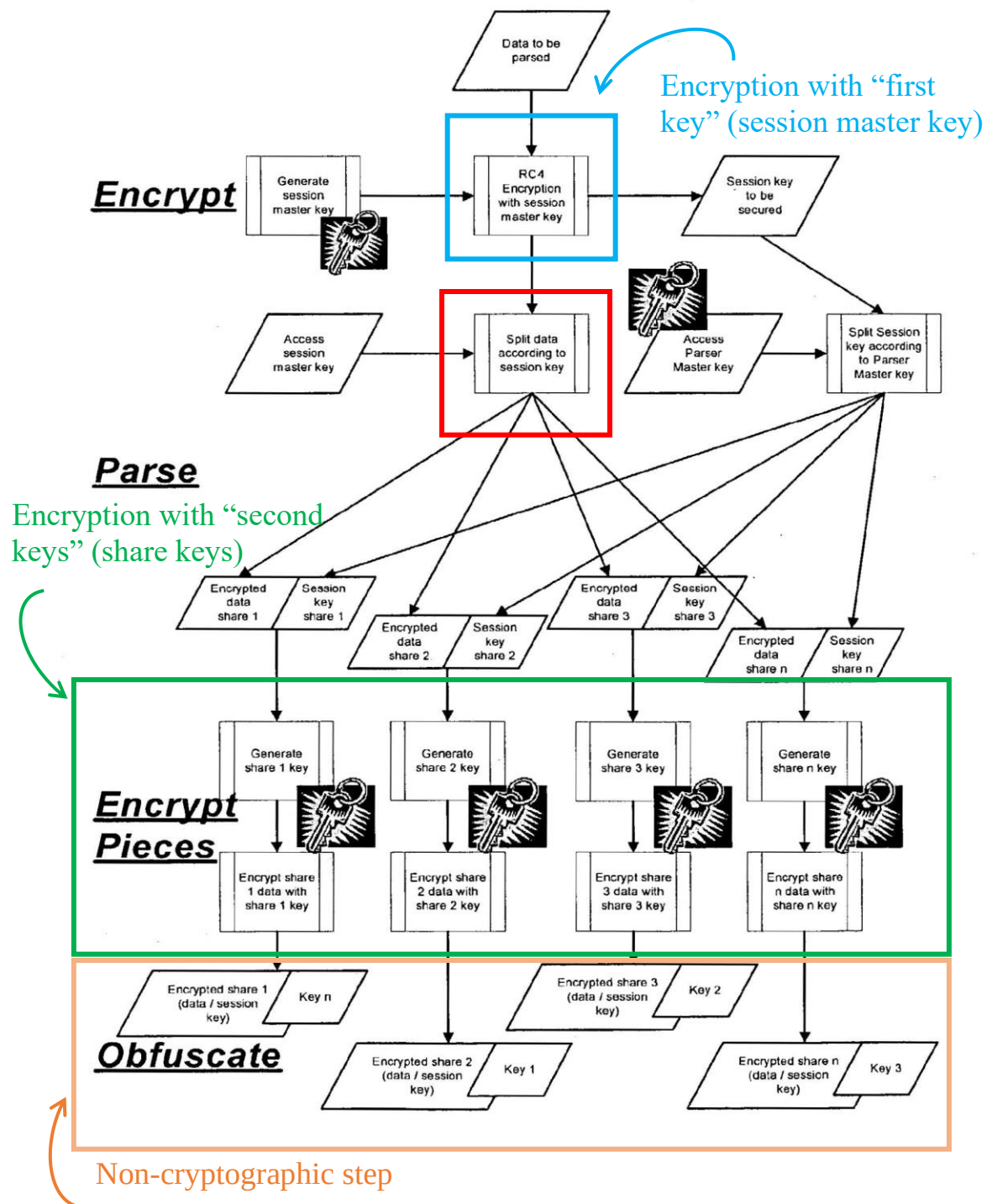
The parties agree that claim 1 requires the generation of data chunks from a data set, and that the data chunks must first be encrypted with a respective second key before the plurality of data chunks is “*further* secur[ed]” with a “*cryptographic* operation based on the first key.” Pet., 15 (“claim 1 recites an encryption step with an encryption key (specifically, a ‘first key’) that occurs after a step corresponding to the ‘encrypt pieces’ stage”), 30-36. There is no dispute that these steps must be performed in the claimed order. Ex. 1064 [Google-Proposed-Claim-Construction] 16 (claim 1 steps “must be performed in order”); *see also Mantech Envtl. Corp. v. Hudson Envtl. Servs.*, 152 F.3d 1368, 1375–76 (Fed. Cir. 1998) (“steps must be performed in order” where “the sequential nature of the claim steps is apparent from the plain meaning of the claim language”). Here, without being already “encrypt[ed] ... with a respective second key,” the plurality of data chunks could not be “*further* secure[d]” with a “cryptographic operation based on the first key.” Ex. 2003 [Rubin-Decl.] ¶ 44.

Petitioner contends that Orsini anticipates claim 1 but fails to show that Orsini expressly or inherently discloses the required ordering.

1. Orsini Does Not Expressly Disclose Claim 1.

Petitioner contends that Orsini's Figure 21 anticipates Claim 1. Pet., 9-16. Orsini, however, does not expressly disclose, "performing a cryptographic operation ***based on the first key*** to ***further*** secure the plurality of data chunks" after generating chunks from a data set and encrypting the chunks. As Dr. Rubin explains:

Assuming, as Petitioner does (*e.g.*, Pet., 11-12, 18, 30-31) that Orsini's "session master key" corresponds to the claimed "first key," Figure 21 undisputedly teaches **a cryptographic operation performed on the entire data set with the session master key** (a "first key") **before parsing the data set and session master key into shares** and **encrypting those shares and split session master key with a set of share keys**. And any additional protection of the data chunks is achieved through a **non-cryptographic step**:

Figure 21

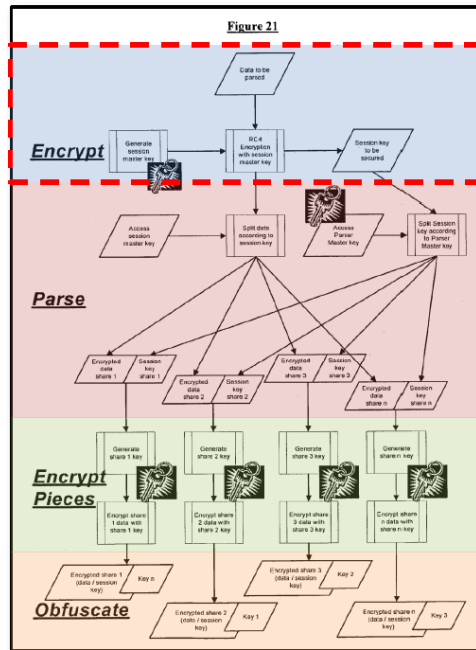
Ex. 1006 [Foster] Fig. 21 (annotated).

Orsini, however, does not disclose performing any “cryptographic operation . . . to further secure the plurality of data chunks” “based on the first [session master] key.” Indeed, Orsini ***cannot disclose this***, because in Figure 21, after the session master key/first key is used to encrypt the dataset as a whole, it is “separat[ed]” along with the data set and those “portions of the session master key” are “contain[ed]” in the chunks/shares of data which are then encrypted. Ex. 1005 [Orsini] [0293], [0297]-[0299]; *compare id. with* Ex. 1001 [’609] 80:24-27 (using workgroup key instead of session master key). Consequently, by the time the data set has been parsed into shares and those shares have been encrypted, the session master key (*i.e.*, the alleged “first key”) cannot be used for a cryptographic operation without removing the benefits provided from Orsini’s splitting and encrypting of that key.

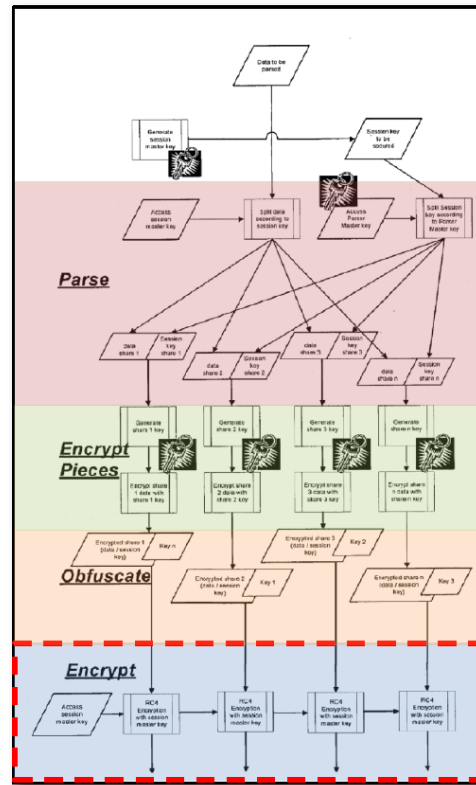
Ex. 2003 [Rubin-Decl.] ¶¶ 47-48.

Rather than pointing to any actual disclosure in Orsini, Petitioner is forced to ***alter*** Figure 21 to add another, undisclosed, chunk encryption step, this time using the session master key, after the data chunks are generated and encrypted a first time with respective second keys:

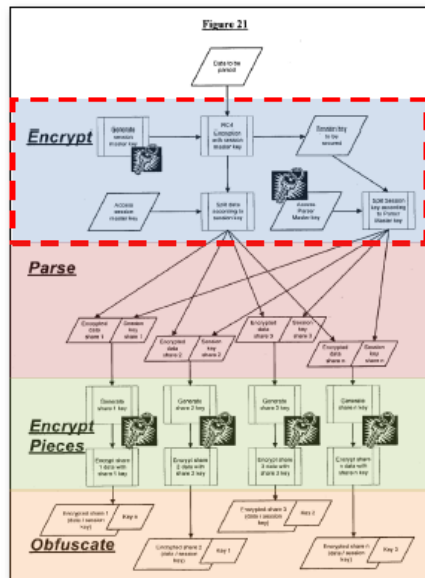
Original



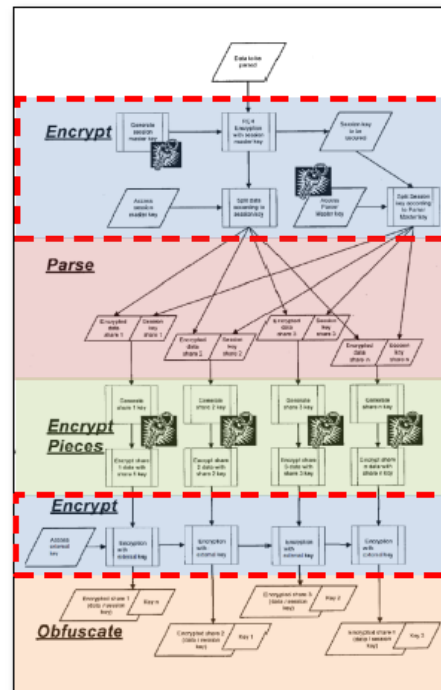
Different-Order Version



Original



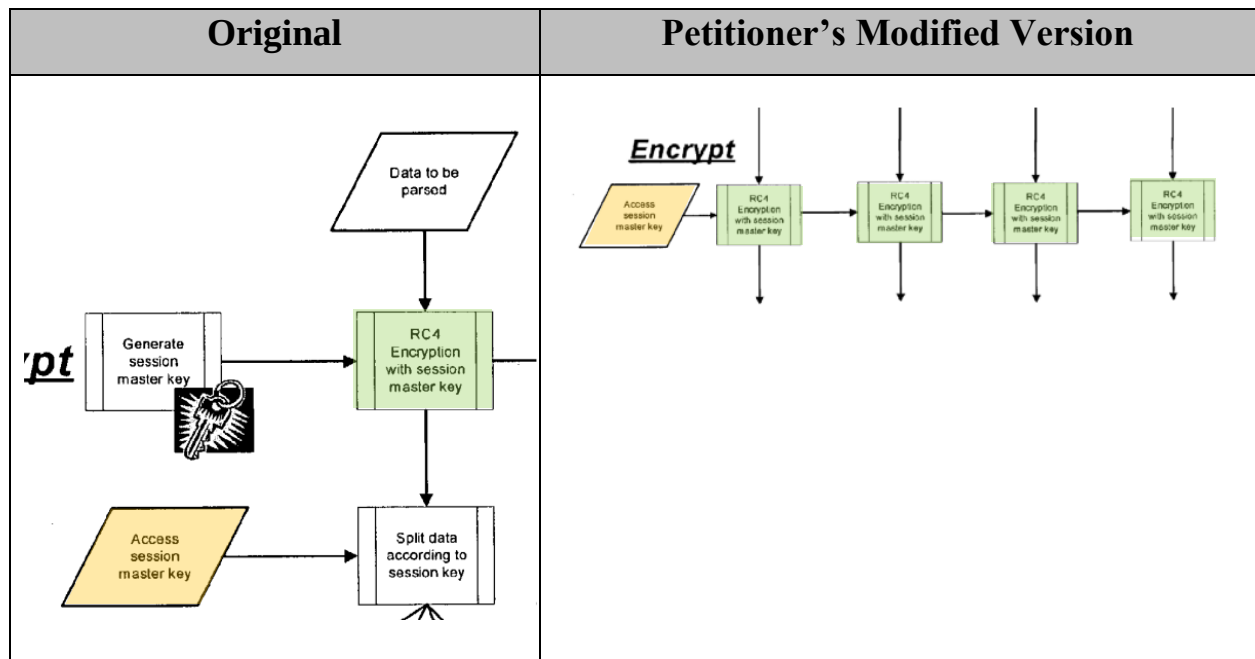
Two "Encrypt" Phases (Version 1)



Pet., 16, 35; *see also id.*, 32-33, 36.

As shown below, not only does Petitioner move (or copy and move) the dataset encryption step from the beginning of Figure 21 to the end, but it also then re-writes the step **to apply to each of the chunks rather than to the dataset as a whole**. Thus, Petitioner does not merely reorder Figure 21, it fundamentally alters what is being encrypted (the dataset versus the shares). Petitioner additionally moves the step **wherein the session master key is accessed from prior to the data being parsed to after the non-cryptographic step**. Ex. 2003 [Rubin-Decl.]

¶ 50.



Petitioner and Dr. Bhattacharjee's reliance upon materially altered versions of Figure 21 confirms, as a matter of law, that Orsini does not anticipate Claim 1.

Enplas Display Device Corp. v. Seoul Semiconductor Co., Ltd., 909 F.3d 398, 405 (Fed. Cir. 2018) (“Prior art that must be modified to meet the disputed claim limitation does not anticipate the claim.”).

2. Orsini Does Not Inherently Disclose Claim 1.

Bereft of an express disclosure of the “further securing” step, Petitioner, nonetheless, contends that Orsini discloses the claimed ordering, relying upon Orsini’s statement that “[i]t is readily apparent to those of ordinary skill in the art that certain steps of the methods described herein may be performed in different order, or repeated multiple times, as desired.” Pet., 30-36 (citing Ex. 1005 [Orsini] [0301]). But, such boilerplate language is not sufficient to show actual disclosure of a claim limitation, for “[a] [prior art] patent discloses only that which it describes” *In re Benno*, 768 F.2d 1340, 1346 (Fed. Cir. 1985) (reversing obviousness rejection); cf. *Forum US, Inc. v. Flow Valve, Ltd. Liab. Co.*, 926 F.3d 1346, 1352 (Fed. Cir. 2019) (“the boilerplate language that modifications can be made to the original disclosed invention does not even suggest an arbor-less embodiment of the disclosed invention.”); *D Three Enters., LLC v. SunModo Corp.*, 890 F.3d 1042, 1050-51 (Fed. Cir. 2018) (“boilerplate language . . . is not sufficient to show adequate disclosure of the actual combinations and attachments used in the Washerless Claims.”).

Equally important, Petitioner does not, and cannot, point to any step in Orsini that could be “performed in a different order” or “repeated” to achieve the limitations of claim 1. No matter how many times one repeats the step of encrypting ***the original data set*** using the session master key, and regardless of the order in which that step is carried out, it cannot disclose encrypting ***the chunks***, as Petitioner contends is a way of meeting the “further securing” step of claim 1. As Dr. Rubin explains:

The claims require generating data chunks from the data set (1[c][1]), then encrypting each of those data chunks with respective second keys (1[d]), and only then “performing a cryptographic operation based on the first key to further secure the plurality of data chunks” (1[e]). Thus, Orsini cannot satisfy claim 1 merely by reordering an encryption step from before splitting the data set to after. Indeed, Orsini’s Figure 21 already teaches encrypting data chunks after the data set is split with respective second keys. Ex. 1005 [Orsini] [0299] (Step “5. Encrypting each share”). Rather, the claims require another cryptographic operation based on the first key to “***further*** secure” the data chunks ***after*** those data chunks are already encrypted.

Ex. 2003 [Rubin-Decl.] ¶ 51; *see also* ¶ 54.

Tacitly acknowledging the absence of a disclosure, Petitioner tries to convert the statement into an inherent disclosure of the limitation, by arguing that POSITAs “understood” that its proposed alterations of Figure 21 were taught by

Orsini. Petitioner claims that “POSAs understood that one ‘certain step[]’ capable of being reordered was the first ‘encrypt’ step with the session master key in Orsini’s Figure 21 [] because it was well-known to reorder whether an encryption step is performed before and/or after splitting.” Pet., 30-31 (citing Ex. 1003 [Bhattacharjee-Decl.] ¶¶ 102-103).

An anticipation argument based on what POSITA might “understand” must meet the requirements for an inherent disclosure. *Microsoft Corp. v. Biscotti, Inc.*, 878 F.3d 1052, 1065 (Fed. Cir. 2017) (statements that “a [POSITA] would understand the [evidence] as disclosing [the limitation]” could support an argument that a limitation was “inherently disclose[d] ...”); *LBT IP I LLC v. Apple Inc.*, 2023 U.S. App. LEXIS 14376, *8 (Fed. Cir. 2023) (similar). This requires a showing that the limitation is “**necessarily** present” in the allegedly anticipating reference. *HTC Corp. v. Cellular Commns. Equip., LLC*, 877 F.3d 1361, 1368-69 (Fed. Cir. 2017).

Petitioner, however, fails to show that this statement in Orsini “necessarily” discloses performing a cryptographic operation to further secure the chunks based on the first/session master key, as required by claim 1. Petitioner provides only a conclusory statement with a single citation to its expert declaration, no explanation or analysis, and a string of unexplained citations in its expert’s declaration, none of which is sufficient to show anticipation. See Pet., 30-31; see also 37 C.F.R.

§ 42.6(a)(3); *Schumer v. Lab. Computer Sys., Inc.*, 308 F.3d 1304, 1315-16 (Fed. Cir. 2002) (“testimony [concerning anticipation] is insufficient if it is merely conclusory.”).

But even if it were considered, Orsini’s boilerplate statement that “certain steps may be performed in different order, or repeated multiple times” does not remotely ***necessarily disclose*** “performing a cryptographic operation based on the first key to ***further secure the plurality of data chunks.***” That “certain” steps can be repeated, would not necessarily indicate that the chunks, each encrypted with a separate share key, would be again encrypted using the session master key (the cryptographic operation Petitioner contends would be reordered/repeated). As Dr. Rubin explains:

Petitioner and Dr. Bhattacharjee rely upon Orsini’s statement that “certain steps of the methods described herein may be performed in different order, or repeated multiple times, as desired.” Pet., 30-36 (citing Ex. 1005 [Orsini] [0301]). This statement, however, does not necessarily disclose that the parsed data would be further encrypted using the first/session master key as Petitioner argues it does. Indeed, neither parsed data nor the session master key is even mentioned in this statement.

Orsini, moreover, only specifies that “***certain*** steps” “may be performed in different order.” Ex. 1005 [Orsini] [0301]. Orsini does not teach that the encryption of the data set step is one such step.

Rather, Orsini discloses that “[f]or example, multiple parsing steps may be performed on only one portion of the parsed data,” and that “[e]ach portion of parsed data may be uniquely secured.” *Id.* Orsini, therefore, only identifies the parsing and encryption with the set of second keys as steps which may be repeated or reordered, and on “only one portion of the parsed data” at a time at that. There is simply no disclosure that would necessitate that the data chunks, after already being individually encrypted with the alleged set of second keys, would be encrypted again, let alone with the alleged first key, or that the “plurality of data chunks” rather than an individual data chunk would be encrypted based on the first key. Indeed, a POSITA would recognize that encryption with the alleged first key would not be an appropriate step to repeat or reorder because it is parsed along with the data set into portions and those portions are then encrypted. Consequently, encrypting the shares with the alleged first key (*i.e.*, the session master key) would render the splitting and encryption of the alleged first key pointless.

Ex. 2003 [Rubin-Decl.] ¶¶ 52-53.

The references Petitioner’s expert cites for the generic proposition that it was known to encrypt data before and/or after fragmenting a dataset are simply irrelevant as to what **Orsini** necessarily discloses. Indeed, Orsini teaches that a dataset may be encrypted before and after parsing, but it does not disclose “further securing” “data chunks” via a “cryptographic operation” based on a “first key.” *See* Ex. 1003 [Bhattacharjee-Decl.] ¶ 103 (citing Exs. 1028, 1053, and 1045).

Finally, Petitioner also points to certain dependent claims in Orsini that claim encrypting shares/chunks a second time using “a different encryption algorithm.” Pet., 33 (citing Ex. 1001 [’609] cls. 12, 29, 39). But these claims do not disclose claim 1’s limitations. As Dr. Rubin explains:

Dependent claims 12, 29, and 39, directed to repeating encryption of shares with a “different encryption algorithm,” do not disclose claim 1, which requires that the encrypted data shares undergo a “cryptographic operation ***based on the first key*** to further secure the plurality of data chunks.” Simply repeating encryption using a “different algorithm” does not even suggest, much less necessarily disclose, that the “different algorithm” would be based on the session master key. See Ex. 1005 [Orsini] [0071] (explaining “a wide number of algorithms [are] available”).

Ex. 2003 [Rubin-Decl.] ¶ 56.

In any event, the dependent claims cited are different embodiments from Figure 21, and Petitioner may not “mix and match” features from different unconnected embodiments to anticipate claim 1. “‘It is not enough that the prior art reference . . . includes multiple, distinct teachings that the artisan might somehow combine to achieve the claimed invention’ to show anticipation.”

BlephEx, LLC v. Myco Indus., 24 F.4th 1391, 1400 (Fed. Cir. 2022); *Net MoneyIN, Inc. v. VeriSign, Inc.*, 545 F.3d 1359, 1371 (Fed. Cir. 2008) (anticipation requires “the limitations [be] arranged or combined in the same way as recited in the

claim”); *Finisar Corp. v. DirecTV Group, Inc.*, 523 F.3d 1323, 1334 (Fed. Cir. 2008) (similar). As Dr. Rubin explains:

None of the claims that Petitioner cites claim Orsini’s Figure 21 embodiment. Again, Orsini’s Figure 21 describes encrypting a data set with a session master key, splitting the data set and session master key into shares, encrypting those shares with respective share keys, and then storing the shares with a share key not used to encrypt it. Ex. 1005 [Orsini] Fig. 21, [0295]-[0299]. The claims Petitioner cites, however, are each far afield of Figure 21. For example, claim 10, from which claim 12 depends, does not even involve a master key or another set of keys with which to encrypt the data or a step hiding the share keys with different shares. ... Similarly, claims 20 and 30 (from which claims 29 and 39 depend) do not require generating a key for each of the data portions and are also missing the step hiding the share keys. Thus, the dependent claims Petitioner cites are not even relevant to what Figure 21 may or may not disclose.

Ex. 2003 [Rubin-Decl.] ¶ 57. Quite simply, these unrelated, dependent claims cannot show that the Figure 21 embodiment discloses claim 1’s invention.

C. Petitioner Fails To Show That Orsini Renders The Claims Obvious.

Petitioner alternatively contends that Claim 1 is obvious over Orsini alone, relying heavily upon a POSITA’s common sense. Petitioner’s argument fails as a matter of law, however, as the Federal Circuit has made clear that a missing limitation may not be satisfied by mere common sense. *See* Section II.C.1. Even

if that could be overlooked, Petitioner fails to demonstrate a POSITA would be motivated to modify Orsini as proposed. *See* Section II.C.2.

1. Petitioner Cannot Rely On Common Sense To Supply Missing Limitation 1[e].

Instead of combining Orsini with another reference, Petitioner contends that, based on her knowledge and common sense, a POSITA “had multiple reasons to implement Orsini to perform” limitation 1[e] (“performing a cryptographic operation based on the first key to further secure the plurality of data chunks”). Pet., 36; *id.*, 36-42 (describing allegedly “known” cryptographic operations and asserting POSAs “had reason” to implement them in Orsini’s Figure 21 method).

The Federal Circuit has held that when a “missing limitation goes to the heart of an invention” a court’s review of any purported “reasoned basis for resort to common sense” must be especially “searching” *Arendi S.A.R.L. v. Apple Inc.*, 832 F.3d 1355, 1362-63 (Fed. Cir. 2016). What is more, “common sense” can be used to supply a missing limitation “**only** when ‘the limitation in question [is] unusually simple and the technology particularly straightforward.’” *DSS Tech. Mgm’t v. Apple Inc.*, 885 F.3d 1367, 1374-75 (Fed. Cir. 2018) (citations omitted).

In this case, the limitation-at-issue is both at the heart of the invention and involves complex technology, so it is not appropriate to supply it through “common sense.” As Dr. Rubin explains:

The inventions-at-issue concern cryptography, which is a technologically-involved and complex field involving incredibly high stakes and safeguarding against a wide variety of highly sophisticated and ever-evolving cybersecurity attacks. As internationally-renowned cryptologist Bruce Schneier (Ex. 2005 [Schneier-CV]) has observed “[f]ew can do the science [of cryptography] properly” and “[g]ood cryptographers know that nothing substitutes for extensive peer review and years of analysis.” Ex. 2006 [Schneier-Why-Cryptography-Is-Harder-Than-It-Looks] 3.

And, as previously discussed, the ’609’s new teachings and the claimed inventions arose from the inventor’s insight regarding a potential vulnerability in Orsini’s Figure 21 approach wherein if an attacker obtains each of the shares, the dataset can be reconstructed. The ’609 eliminates this vulnerability by encrypting the encryption keys using a workgroup key—corresponding to the “first key” of claim 1—a technique not disclosed in Orsini. The workgroup key provides “enhanced protection of information assets by encrypting key information stored within the shares” (Ex. 1001 [’609] 79:60-62) adding “an extra level of security to a secured data set” (*id.*, 77:10-14). As the ’609 explains “even if all required shares and external keys are discovered, an attacker has no hope of recreating the information without access to the workgroup key.” *Id.*, 79:62-65. Thus, this limitation goes to the heart of the invention. Moreover, given the benefits associated with the “workgroup key” delineated in the ’609, it further seems clear that the inventors of the ’609 (and Orsini) considered the “workgroup key” concept to be a fundamentally

different approach that solved a problem that Orsini did not recognize or address.

Ex. 2003 [Rubin-Decl.] ¶¶ 60-61.

Consequently, Petitioner's attempt to rely solely on the common sense of a POSITA to add the missing limitation to Orsini should be rejected here because the missing limitation arises in a technically complex field and presents a fundamentally different approach from Orsini, disclosing and claiming new technology to address a potential vulnerability in Orsini's approach (*see* Section II.A). The limitation-at-issue thus lies at the heart of the claimed inventions.

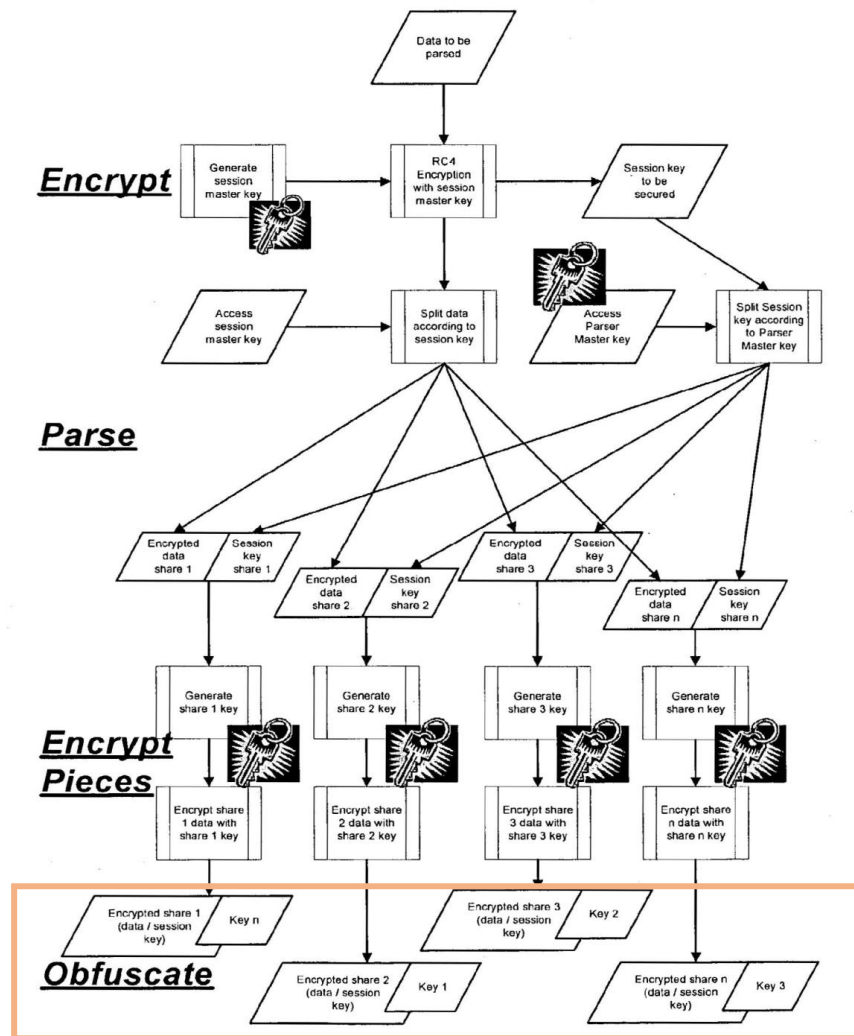
2. Petitioner Fails To Show That A POSITA Would Have Been Motivated To Modify Orsini to Add Limitation 1[e].

As discussed, like claim 1, Orsini includes the steps of generating and encrypting the chunks but does not disclose limitation 1[e]: “performing a cryptographic operation based on the first key to further secure the plurality of data chunks.” *See* Section II.A. Petitioner suggests that “POSAs would have been motivated by [Orsini's] disclosures” to add this limitation because they “would have understood that doing so would have provided additional security and efficiency benefits.” Pet., 36-41. Petitioner, however, fails to demonstrate that a POSITA would be motivated to modify Orsini to “perform[] a cryptographic operation based on the first key to further secure the plurality of data chunks.”

Petitioner cannot show that a POSITA would understand that adding a “cryptographic operation based on a first key would offer “additional security and efficiency benefits” in the Orsini method. Orsini **already** hides the encryption key used to encrypt each data chunk with a different chunk—a **non**-cryptographic operation that is **not based on any key**. Ex. 1005 [Orsini] Fig. 21, [0299]. As Dr. Rubin explains:

Orsini already teaches hiding the encryption key used to encrypt each data chunk with a different chunk through a **non-cryptographic process**, illustrated in Figure 21 below.

Figure 21



Ex. 1005 [Orsini] Fig. 21. As can be seen in Figure 21, in this step, the system “store[s] the encryption keys in different locations from the encrypted data portions or shares ...” *Id.*, [0299]; see also [0353] ...; [0396]; cl. 26.

As Petitioner does not dispute, the step taught in Orsini’s Figure 21 is not a cryptographic operation (much less one involving the first key). See generally *Pet.*; Ex. 1003 [Bhattacharjee-Decl.]. As there is no encryption scheme or “scrambling” or altering of the data or the

alleged respective second keys (Ex. 2007 [Katz] 4), it cannot be a cryptographic operation.

Ex. 2003 [Rubin-Decl.] ¶¶ 63-64.

Petitioner fails to explain why a POSITA would be motivated to modify Orsini using other allegedly “known” methods, particularly when it non-cryptographically hides each share key with a share it did not encrypt. Indeed, Orsini teaches that its method provides a “***comprehensive package for data security.***” Ex. 1005 [Orsini] [0380] (“The data shares are then encrypted individually and the keys are stored safely with different encrypted shares. When combined, the entire process for securing data according to the methods disclosed herein becomes a comprehensive package for data security.”); Ex. 2003 [Rubin-Decl.] ¶ 65.

Petitioner’s contention that a POSITA would be motivated to modify Orsini does not address Orsini’s non-cryptographic step or explain why a POSITA would consider Orsini’s “comprehensive package for data security” insufficient. Instead, Petitioner appears to argue that a motivation to modify Orsini arose because POSITAs “knew” that “cryptographic keys should never occur in clear form outside the cryptographic device” and that “when ‘encryption keys are included with the encrypted message’ ‘[s]ecurity’ comes from ‘encrypt[ing] the encryption keys themselves’ with ‘a key encryption key.’” Pet., 40 (citing Exs. 1035, 1036).

These concerns, however, are addressed by Orsini's non-cryptographic step, by storing the keys, not with the encrypted message but rather "in different locations from the encrypted data portions or shares." Ex. 1005 [Orsini] [0299]. Orsini teaches that this is sufficient because an attacker would not have "enough information to decrypt the" data chunk stored in each data depository. *Id.*, [0353]; Ex. 2003 [Rubin-Decl.] ¶ 66. Notably, in Figure 21's data-securing method, the session master key, data chunks, and session keys are all either encrypted, parsed, and/or hidden, and thus each potentially vulnerable part of the encryption method is secured. Ex. 2003 [Rubin-Decl.] ¶ 66.

Thus, a POSITA would recognize that Orsini already employs a non-cryptographic method rather than a "cryptographic operation based on the first key to further secure the plurality of data chunks" as required by claim 1. It is not obvious to modify a reference that already achieves the benefit supposedly served by the proposed modification. *See, e.g., In re NTP, Inc.*, 654 F.3d 1279, 1289 (Fed. Cir. 2011) (reversing IPR obviousness rejections as "based on improper hindsight reasoning" when "the [prior art] already discloses an RF network that connects portable computers to the system. Thus, adding an RF network [in the way necessary to meet the claims] would render the RF network [in the prior art] superfluous."); *Apple Inc. v. Cellular Comm'ns Equipment, LLC*, IPR2015-00576, Paper 7, 12 (June 12, 2015) (denying institution where "Petitioner ... fails to

provide a specific explanation why [the primary reference] did not already accomplish [the asserted benefits] by itself.”); *F5 Networks, Inc. v. Radware, Ltd.*, IPR2017-00124, Paper 48, 64-65 (Apr. 23, 2018) (no motivation for modification that “serve[s] no purpose”).

Petitioner not only fails to demonstrate that a POSITA would understand that its proposed modifications add any benefit to Orsini, but it also fails to consider their drawbacks. The Federal Circuit has held that “[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.” *Arctic Cat Inc. v. Polaris Indus.*, 795 Fed. App’x. 827, 833 (Fed. Cir. 2019); *Henny Penny Corp. v. Frymaster LLC*, 938 F.3d 1324, 1331-32 (Fed. Cir. 2019) (similar). Petitioner has not even considered that there may be drawbacks let alone perform such weighing. *Arctic Cat Inc. v. Bombardier Rec. Prods.*, 876 F.3d 1350, 1363 (Fed. Cir. 2017) (“The potential problems” with combination coupled with “an alternative approach [that] might reduce those same problems,” supported finding that “a skilled artisan would not have been motivated” to make combination).

Here, the drawbacks of Petitioner’s proposed modification of Orsini are considerable. As Dr. Rubin explains, cryptography is a complicated field where additional encryption steps, such as that proposed by Petitioner, frequently backfire:

Simply concluding that adding an encryption layer would “benefit” Orsini’s method oversimplifies the myriad considerations a POSITA must make in cryptography, even for what may seem to be a “simple” or “small” change. For example, a POSITA would recognize that when fully considering the costs and benefits, an additional layer of encryption may not in reality benefit an encryption scheme or “further secure the plurality of the data chunks” as the claims require. Rather, an additional encryption step may in fact introduce a weakness that an attacker can exploit. *See, e.g.,* Ex. 2007 [Katz] 22 (“As an example, an encryption scheme that has been proven secure ... might be implemented on a smart-card. It may then be possible for an adversary to monitor the power usage of the smartcard (e.g. how this power usage fluctuates over time) and use this information to determine the key.”); Ex. 2008 [Schneier-Security-Pitfalls-in-Cryptography] (“And just as it’s possible to build a weak structure using strong materials, it’s possible to build a weak cryptographic system using strong algorithms and protocols.”); Ex. 2009 [Anderson] (new computer “security systems have a nasty habit of backfiring”). One such weakness is in the critical area of key management, as it is where systems tend to be most vulnerable. *See* Ex. 2010 [Schneier-Applied-Cryptography] 169 (“In the real world, key management is the hardest part of cryptography. Designing secure cryptographic algorithms and protocols isn’t easy, but you can rely on a large body of academic research. Keeping the keys secret is much harder.”).

Thus, while adding an additional round of encryption with the same or additional keys may appear to provide improved security, doing so may, on balance, compromise security.

Ex. 2003 [Rubin-Decl.] ¶¶ 68-69.

Here too, Petitioner's proposal to modify Orsini's Figure 21 to perform a further cryptographic operation with the session master key entails significant risks, which Petitioner's obviousness analysis does not consider. As Dr. Rubin explains, Petitioner's approach may require exposing or undoing the prior security steps in the Orsini method:

Petitioner's proposal to "re-encrypt data chunks with a first encryption key" (Pet., 36-37) has significant security flaws. Ex. 2010 [Schneier-Applied-Cryptography] 357 ("Encrypting a plaintext block twice with the same key, whether with the same algorithm or a different one, is not smart."). If Orsini were modified such that the session master key (the alleged "first key") were either reapplied or otherwise utilized after the data has been split, that would undermine Orsini's method which protects the alleged first key by splitting it amongst the data shares. And it would entail further complications. By involving the session master key after the splitting process, Petitioner's proposal would necessitate the key management and transport of the session master key, effectively making the division and inclusion of the session master key within the encrypted shares pointless. Under Orsini's existing method, the master session key has already been parsed among the data shares and encrypted. Consequently, the session master key

which was previously secured in Figure 21 would be unnecessarily exposed in this proposed modification. “Encrypting...already-encrypted data chunks with the same encryption key,” thus, presents security vulnerabilities and complications that Petitioner and Dr. Bhattacharjee fail to account for.

Ex. 2003 [Rubin-Decl.] ¶ 70.

To the extent Petitioner proposes to modify Orsini’s Figure 21 to use another key, such as the proposed “group key,” “encryption key,” or “key-encryption key” (Pet., 37-41), Petitioner too fails to address the drawbacks associated with managing and securing such keys against discovery by an attacker. Consequently, Petitioner fails to demonstrate that a POSITA would be motivated to implement its proposed key-management solution and thus fails to demonstrate that a POSITA would be motivated to use the proposed “group key,” “encryption key,” or “key-encryption key.” As Dr. Rubin explains:

Petitioner and Dr. Bhattacharjee argue that “Orsini provides ‘**session** key management’” and discloses a “single point of . . . key management” that could be modified to “centrally generat[e] Orsini’s session master keys (*and any of the conventional encryption keys [used in Petitioner’s obviousness argument]*).” Pet., 23-24 (citing Ex. 1005 [Orsini] [0176]-[0179], [0293], [0381]-[0386], [0412]). But **session** key management is very different than key management for longer-lived keys such as the other “conventional encryption keys” Petitioner relies upon, especially as Figure 21’s session keys (the

session master key and share keys) are stored alongside the encrypted data. For example, “session keys should be changed at least daily” if not for each “particular communications session” while “key-encryption keys don’t have to be replaced as frequently,” perhaps “once a year.” Ex. 2010 [Schneier-Applied-Cryptography] 184, 47; *see also id.*, 184 (“Encryption keys used to encrypt data files for storage cannot be changed often.”).

This longer lifespan, however, makes potential compromise of a key-encryption key all the more consequential. *Id.* (“However, if a key-encryption key is compromised, the potential loss is extreme: all communications encrypted with every key encrypted with the key-encryption key.”). Consequently, a POSITA would recognize that a key management solution for session keys would not be appropriate for the “group key,” “encryption key,” or “key-encryption key” used in Petitioner’s alternative proposed modifications. *See* Ex. 1001 [’609] 62:22-23 (parser master, session master, and share encryption “keys may have individual key storage, retrieval, security and recovery options.”).

Moreover, key management for a “group key,” “encryption key,” or “key-encryption key” would necessarily be different than for key management for the “session” keys in Figure 21. In Figure 21, the session master key used to encrypt the dataset is parsed within the data shares (Ex. 1005 [Orsini] [0297]) and the session share keys are stored along with the data shares (*id.*, [0299]). Therefore, so long as the party reconstructing the dataset possesses the shares, he or she also possesses

the session keys necessary to decrypt the data. By contrast, the “group key,” “encryption key,” or “key-encryption key” would not be stored alongside the data but would rather need to be distributed out-of-band to the decrypting party. Consequently, those keys would need to be maintained and stored on both ends and in transit such that no adversary gets access to them. Thus, the threat model for the “group key,” “encryption key,” or “key-encryption key” is very different than for a “session key.” Petitioner and Dr. Bhattacharjee have not mentioned, let alone adequately addressed, these different threat models. Their superficial consideration of key management, an incredibly important aspect of cryptography, is thus insufficient and flawed.

Ex. 2003 [Rubin-Decl.] ¶¶ 72-74.

Instead of weighing the benefits and drawbacks of its proposed modifications, Petitioner attempts to justify them by claiming they are the application of a “known” method, citing a number of other references not included in Ground 1. Pet., 36-42. Although such external references may “document the knowledge that skilled artisans would bring to bear in reading the prior art identified as producing obviousness,” *Ariosa Diagnostics v. Verinata Health, Inc.*, 805 F.3d 1359, 1365 (Fed. Cir. 2015), Petitioner cannot subvert the requirement to properly identify the relied-on combinations of references in a Petition by citing them as documentation of background knowledge, when it is actually using them to supply a missing limitation. Pet., 37-41. And it cannot use references to

establish a motivation to modify Orsini. *RPX Corp. v. Parity Networks, LLC*, IPR2018-00097, Paper 7, 14 (Apr. 24, 2018) (“Petitioner also cites to several other references . . . but does not assert these references are part of the combination. Pet. 34–35. Relying on knowledge in the art does not . . . show a motivation to combine existing components in the manner claimed free of hindsight motivation.”) (denying institution); *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); *Parsons Xtreme Golf, LLC v. Taylor Made Golf Co., Inc.*, IPR2018-00518, Paper 14, 18 n.4 (July 18, 2018) (similar).¹

In the end, Petitioner’s purported motivation to modify Orsini is entirely hindsight-driven, based on knowledge of the ’609’s disclosure. *See* Section II.A. It is well-settled that “a patent composed of several elements is not proved obvious

¹ Petitioner cites *Intel Corp. v. PACT XPP Schweiz AG*, 61 F.4th 1373, 1380 (Fed. Cir. 2023) for the proposition that “if there’s a known technique to address a known problem using prior art elements according to their established functions, then there is a **motivation to combine**,” but, it is inapposite because Petitioner does not even attempt to show a motivation to combine Orsini with any of these references.

merely by demonstrating,” as Petitioner tries to do here, “that each of its elements was, independently, known in the prior art.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418–19 (2007). Rather, Petitioner must “identify **a reason** that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does.” *Id.* And this it has failed to do. As Dr. Rubin explains:

Petitioner’s proffered modifications do not seem to create any advantage for Orsini’s system and, instead, would seem to undermine Orsini’s approach. At the very least, Petitioner’s proffered modifications are inadequately developed. Instead, Petitioner’s modifications ultimately seem to be driven by hindsight. To reiterate, in each instance, Petitioner begins with Orsini’s Figure 21 which sets out a fully functional system (Ex. 1005 [Orsini] [0380]) that Petitioner has not improved upon. Adopting the solution of “performing a cryptographic operation based on the first key to further secure the plurality of data chunks” seems driven by hindsight, not any technological need or benefit that would improve upon Orsini. It seems the claim limitation itself is being used as a roadmap driving the modification.

Ex. 2003 [Rubin-Decl.] ¶ 75.

Petitioner’s proffered modifications thus provide no benefit, entail significant drawbacks, and are precisely the sort of hindsight-driven analysis the Federal Circuit has rejected. *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.”).

Petitioner’s invocation of using a “group key” or encrypting the “respective second keys” with a “key-encryption key” are likewise hindsight-driven, merely following the ’609’s newly-disclosed inventive concept of encrypting the share encryption keys with a workgroup key. *Compare* Pet., 37-41 with Ex. 1001 [’609] 79:46-65. As the Federal Circuit has held, however, “[t]he inventor’s own path itself never leads to a conclusion of obviousness; that is hindsight.” *Univ. of Strathclyde v. Clear-Vu Lighting LLC*, 17 F.4th 155, 165 (Fed. Cir. 2021) (quoting *Otsuka Pharm. Co., v. Sandoz, Inc.*, 678 F.3d 1280, 1296 (Fed. Cir. 2012)); *InTouch Techs., Inc. v. VGo Commc’ns., Inc.*, 751 F.3d 1327, 1351 (Fed. Cir. 2014) (rejecting expert’s obviousness theory that “relied on the [] patent itself as her roadmap for putting what she referred to as pieces of a ‘jigsaw puzzle’ together.”); *In re Conrad*, 759 Fed. App’x. 982, 985 –86 (Fed. Cir. 2019) (obviousness requires “a specific” rationale to combine limitations to make the claimed invention that is “**unrelated to** the rationale behind the **inventor’s** reasons for making the **[claimed]** invention.”) (collecting cases).

Petitioner, thus, contends a POSITA would be motivated to modify Orsini in a fashion that would yield no benefit identified in the prior art (in view of Orsini storing share keys with shares they were not used to encrypt, which Petitioner ignores), entails significant drawbacks, is based upon inchoate and nonasserted combinations, and is ultimately driven by using the inventor's own path against him coupled with using the claims as a roadmap. Petitioner's proffered modification is rank hindsight and should be rejected.

III. THE FOSTER GROUND FAILS (ALL CLAIMS, GROUND 2).

Petitioner's second ground argues the claims are anticipated and/or rendered obvious by Foster. Pet., 3. Foster, however, is a fundamentally different invention from the '609 and concerns protecting content streamed or transmitted from a content owner to a consumer rather than stored data as the '609 does. *See* Section III.A. Not surprisingly, Petitioner's second ground also fails and for multiple reasons. *See* Sections III.B-E.

A. The '609 And Foster Fundamentally Differ.

The '609 and Foster arise in entirely different contexts, are directed towards addressing fundamentally different problems, and result in very different inventions. As Dr. Rubin explains:

The '609's claimed inventions, and related disclosure, concern securing data within a system. Ex. 1001 ['609] 1:34-39 ("computer

systems... are often storing, archiving, and transmitting all types of sensitive information. Thus, an ever-increasing need exists for ensuring data stored and transmitted over these systems cannot be read or otherwise compromised.”). This is accomplished by the claimed method of parsing the dataset into chunks, encrypting those chunks with respective second keys, further securing the chunks with a cryptographic operation based on a first key, and then storing at least one chunk on a storage device. *Id.*, cl. 1. Each of these steps is intended to secure the dataset. “[C]ryptographic parsing and splitting” of the dataset into chunks and “physical[] remov[al]” of at least one of the encrypted and “further secure[d]” chunks from the system onto a storage device removes the possibility of “attack” or “compromise” of the whole dataset. *Id.*, 56:44-58.

Foster, by contrast, is directed towards a system wherein a content owner (*e.g.*, a movie studio) provides content to Foster’s system for encryption so that the content can be streamed (*e.g.*, via the internet) to a recipient.

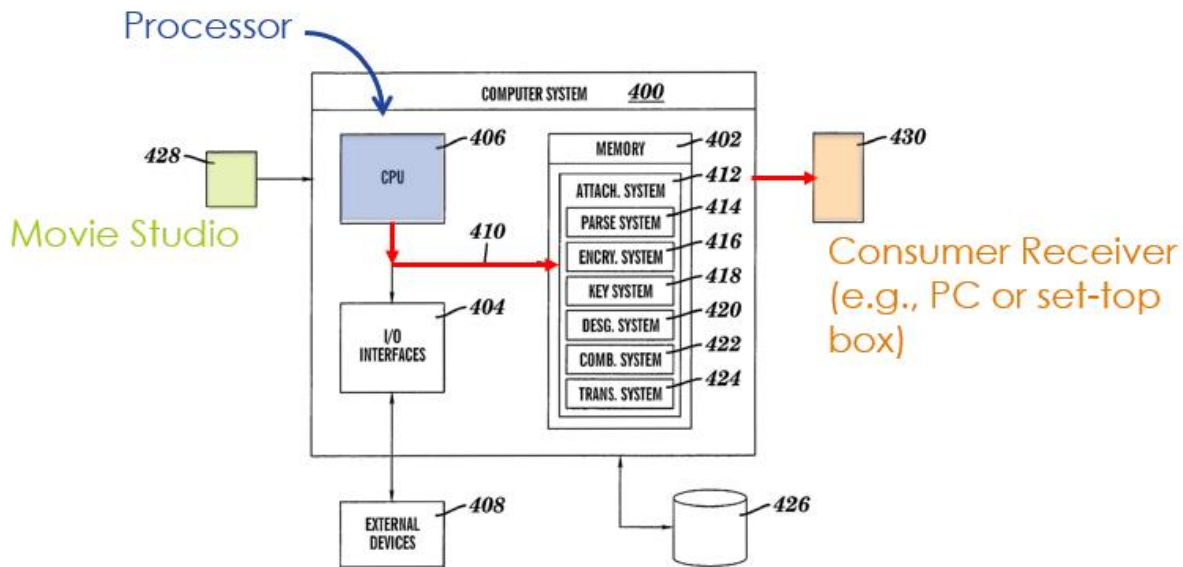


FIG. 16

Ex. 1006 [Foster] Fig. 16 (annotated); [0002] (“a method, system and program product for attaching a title key to encrypted content for synchronized transmission to a recipient.”); [0078] (“computer system 400 is intended to be representative of any system capable of processing content units of an elementary content stream received from source 428, and transmitting the processed content units to a recipient 430.”).

Foster, thus, essentially acts as an encryption middleman for streamed content. Foster, in contrast to the claimed inventions, does not “generate” a plurality of data chunks from a dataset. Rather, it just streams encrypted content in network packets (in accord with the applicable standards). Nor does Foster’s system even store at least one chunk on a storage device.

Ex. 2003 [Rubin-Decl.] ¶¶ 78-80.

Thus, unsurprisingly, Petitioner’s Foster grounds fail for the multiple reasons below.

B. Petitioner Fails To Show Foster Discloses “Generating A Plurality Of Data Chunks Based On A Data Set” (Limitation 1[c][1]).

Limitation 1[c][1] requires “generating a plurality of data chunks based on a data set.” Petitioner maps the claimed “plurality of data chunks” to Foster’s content units (more specifically, their payloads) and the claimed “data set” to Foster’s “content.” Pet., 65-66. Petitioner, however, fails to show that Foster’s “content” is a “data set” or that its “content units” are “data chunks.”

As an initial matter, the parties have agreed that a “data set” must be “*for storage.*” Ex. 2020 [Google-Responsive-Claim-Construction-Brief] 8 n.3. Petitioner simply asserts that Foster’s content meets this construction without any further explanation. Pet., 65-66. But Foster’s definition of “content,” which Petitioner cites, makes no mention of “storage.” Ex. 1006 [Foster] [0008] (“Content—any data such as digital image, sound or binary data deliverable from a source to a recipient”). Petitioner, thus, fails to demonstrate that Foster’s “content” is “a collection of information *for storage*” and consequently fails to demonstrate that Foster teaches a “data set” under the parties’ agreed-upon construction.

Even if, *arguendo*, Foster’s content were a “data set,” Petitioner still fails to demonstrate that Foster teaches “data chunks.” Petitioner asserts with no further explanation that Foster’s “payloads” are “data chunks”:

As illustrated in Foster’s Figure 8 . . . , Foster generates a plurality of “**payloads**” (“**data chunks**”) by “parcel[ing]” the elementary digital content. *See also supra* §VII.B.1 (1[PRE]); Foster, [0064]; Bhattacharjee, ¶¶183-84.

Pet., 65 (emphasis in original). As Dr. Rubin explains, Petitioner’s alleged evidence does not substantiate Petitioner’s assertion:

Petitioner cites its discussion of claim 1’s preamble as support, but that discussion does not mention “payloads” or “data chunks,” let alone explain what “data chunks” are or why Foster’s payloads are “data chunks.” Pet., 60-61. Petitioner also cites Foster at [0064], but this disclosure simply discloses “parcel[ing]” content and does not explain why Foster’s payloads are the claimed “data chunks.” Ex. 1006 [Foster] [0064]. Finally, Petitioner cites Dr. Bhattacharjee’s declaration, but it provides no further insight as it merely parrots the Petition. Ex. 1003 [Bhattacharjee-Decl.] ¶¶ 183-84.

Ex. 2003 [Rubin-Decl.] ¶ 83. Dr. Bhattacharjee’s conclusory testimony should be afforded little or no weight as it is precisely the sort of conclusory testimony rejected in *Xerox. Xerox Corp. v. Bytemark, Inc.*, IPR2022-00624, Paper 9, 15 (Aug. 24, 2022) (precedential) (“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” when “it merely repeats, *verbatim*, the conclusory assertion for which it is offered to support.”).

In fact, the specifications of Foster and the '609 indicate to a POSITA that Foster's “payloads” are not the claimed “data chunks.” As Dr. Rubin explains, Foster's “payloads” are payloads of packets that arise as a result of transmission according to the MPEG video compression standard. The '609, however, teaches that these packets can be secured by parsing them into “data chunks” and therefore are not the claimed “data chunks”:

Petitioner and Dr. Bhattacharjee rely on Foster's “payloads” as the claimed data chunks. Pet., 65. Foster teaches that these “payloads” refer to the payloads of MPEG Packetized Elementary Stream (PES) packets (also referred to in Foster as “units”), as illustrated below in Figure 13:

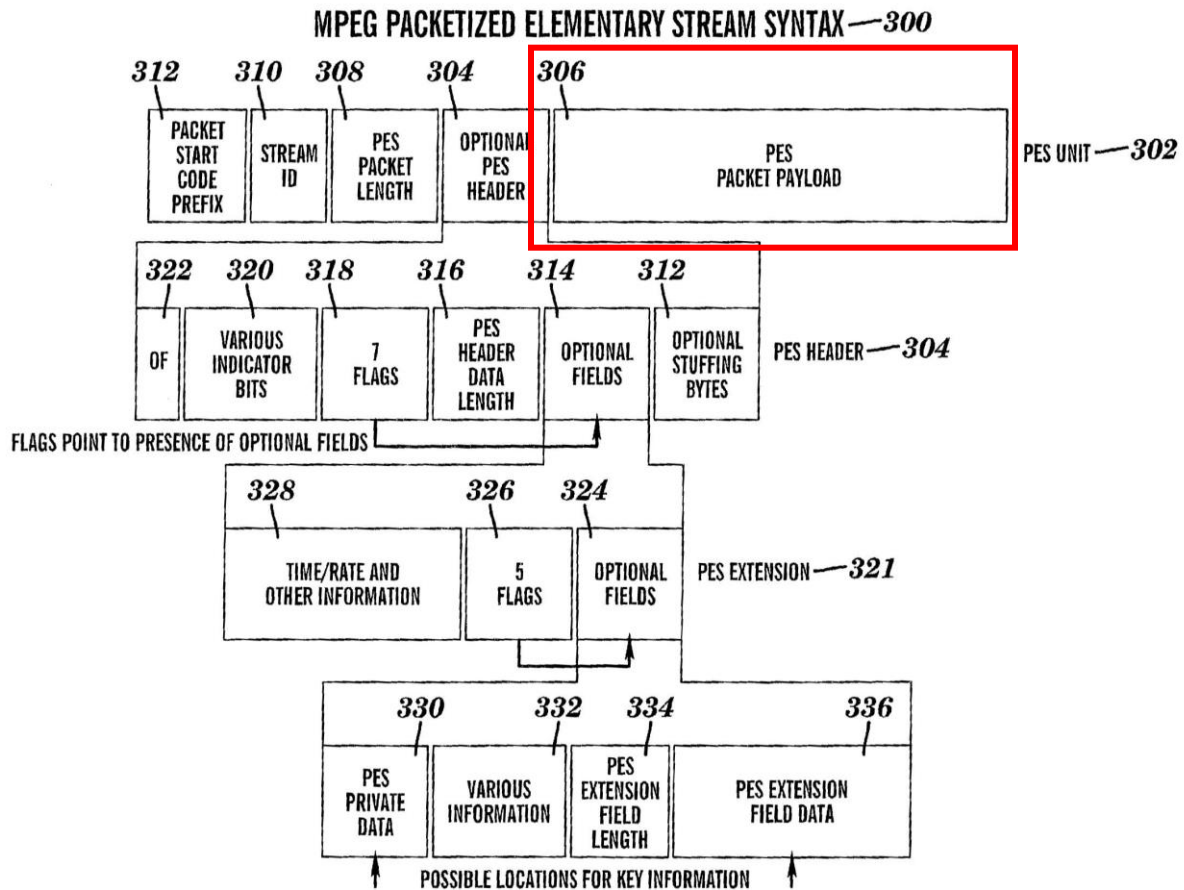


FIG. 13

Ex. 1006 [Foster] Fig. 13, [0072] (“a MPEG packetized elementary stream 300 is parceled into PES (packet) units 302 PES unit 302 includes packetized elementary stream (PES) packet payload 306” which “will be encrypted with a title key”). Foster’s PES packets are also referred to as “data packets.” See Ex. 2011 [Tsai] [0001], cl. 1. Consequently, Petitioner and Dr. Bhattacharjee map the claimed “data chunks” to Foster’s packets (specifically, the payloads of those packets) which are then individually encrypted.

The '609, however, does not refer to data packets as “data chunks.” Rather, the '609 confirms that data packets are themselves parsed into data chunks or shares. Ex. 1001 ['609] 82:15-18 (“At step 4106, *the data packet or data block may then be partitioned into two or more data shares . . .*”), 81:30-31 (“*Packets of data (e.g., network packets) may also be efficiently split* and restored”), 81:48-52 (“...simplified *data partitioning process 4010 ... may be performed on network packets, . . .*”). Thus, the '609 identifies data packets for streaming, such as those disclosed in Foster, as the unit to be parsed, not as “data chunks [generated] based on a data set.” In other words, by disclosing a method for secure transmission of MPEG data packets, Foster does not disclose “generating a plurality of data chunks” as a POSITA would understand the limitation based on the '609's specification. Notably, neither Petitioner nor Dr. Bhattacharjee identify any teaching in the '609 or in the technical literature suggesting that a data packet would qualify as a “data chunk.” By contrast, the literature does discuss packets being delivered in “chunks.” Ex. 2012 [Pek] 3 (“[The d]ata packet is delivered as a sequence of chunks.”); Ex. 2013 [Groote] 3 (“the input is a large data packet ... which is delivered in small chunks.”).

Indeed, the difference between the '609's data chunks and Foster's data packets goes to the fundamental difference between the '609 and Foster's inventions. As the '609 explains, partitioning the data set into chunks increases security:

Traditional encryption technologies known in the art rely on one or more key used to encrypt the data and render it unusable without the key. ***The data, however, remains whole and intact and subject to attack.*** The secure data parser of the present invention, in one embodiment, ***addresses this problem by performing a cryptographic parsing and splitting of the encrypted file*** into two or more portions or shares, and in another embodiment, preferably four or more shares, adding another layer of encryption to each share of the data, then storing the shares in different physical and/or logical locations. When one or more data shares are physically removed from the system, either by using a removable device, such as a data storage device, or by placing the share under another party's control, ***any possibility of compromise of secured data is effectively removed.***

Ex. 1001 ['609] 56:44-58. Thus, the '609 teaches that its data chunks are “generat[ed]” to secure the data.

By contrast, Foster teaches that its content is “parceled” or “packetized” by virtue of the MPEG packetized elementary stream standard. Ex. 1006 [Foster] [0072]-[0073]; Ex. 2014 [Overview-of-MPEG-2-Systems] (“An elementary stream is broken up into packets, forming a *packetized elementary stream* (PES).”). These packets allow for transmission of the content. See Ex. 2015 [Newton's-Telecom-Dictionary] 626 (“**Packet** 1. Generic term for a bundle of data, usually

in binary form, organized in a specific way *for transmission.*”); Ex. 2016 [Oxford-Dictionary-of-Computing] 257 (“a group of binary digits forming a message. The group is *transmitted* together”). And that Foster’s packets are for transmission is of no surprise as Foster’s invention “relates to . . . encrypt[ing] content for synchronized transmission to a recipient.” Ex. 1006 [Foster] [0002]. The ’609 data chunks and Foster’s packets serve entirely different purposes.

Ex. 2003 [Rubin-Decl.] ¶¶ 85-88.

Thus, Petitioner fails to demonstrate that Foster discloses or renders obvious “generating a plurality of data chunks based on a data set.”

C. Petitioner Fails To Show That Foster Renders Obvious “The Data Set Can Be Reconstructed Using At Least A Minimum Number Of The Plurality Of Chunks” (Limitation 1[c][2]).

Limitation 1[c][2] requires “generating a plurality of data chunks based on a data set” such that “the data set can be reconstructed using at least a minimum number of the plurality of chunks.” Ex. 1001 [’609] cl. 1. Under “Google’s proposed construction[,],” “a minimum number of the plurality of data chunks” means “a *predefined* number of the lowest amount of the plurality of data chunks.” Pet., 67. Google’s proposed construction here is the same Petitioner proposes in district court. Ex. 1064 [Google’s-Proposed-Claim-Construction] 13; Ex. 2017 [Google’s-Opening-Claim-Construction-Brief] 22.

Petitioner further contends limitation 1[c][2] is “obvious under Google’s proposed constructions.” Pet., 67. Foster, however, does not teach reconstructing the data set based upon a “**predefined** number of the lowest amount of the plurality of data chunks.” Rather, as Petitioner recognizes, “**all** of [Foster’s] encrypted content packets ... are required in order to access ... the entire stream.” Pet., 67; Ex. 1006 [Foster] [0071] (“because each content packet can be encrypted with a different title key, a party who illicitly obtains a title key cannot access the entire stream”). In Foster, there is simply no “predefined number of the lowest amount of the plurality of data chunks” used to reconstruct the data set. See Ex. 1006 [Foster] [0071] (“the content stream can be accessed at any packet.”).

Petitioner, nonetheless, contends a POSITA “had reason” to modify Foster to include a “known data-distribution technique” (citing Orsini, Birrell, and Cochran) to include redundant data in each data packet so a data set may be recovered using “fewer than all [data packets].” Pet., 67-68. Petitioner further contends “[t]his implementation of Foster’s system would have been a conventional way to distribute and store data” with “known benefits – like redundancy and protection against loss of/inability to retrieve individual data chunks.” *Id.*

Petitioner, thus, cites Orsini, Birrell and Cochran, as evidence that including redundant data in each data packet so the data set could be recovered using “fewer

than all” data packets was a known technique. Pet., 67-68 (citing Exs. 1005 [Orsini] [0280]-[0284]; 1050 [Birrell]; 1052 [Cochran]). But none of Orsini, Birrell, or Cochran are a part of Petitioner’s combination and, consequently, they should not even be considered as a basis for modifying Foster. *RPX*, IPR2018-00097, Paper 7, 14 (“Petitioner also cites to several other references . . . but does not assert these references are part of the combination. Pet. 34–35. Relying on knowledge in the art does not . . . show a motivation to combine existing components in the manner claimed free of hindsight motivation.”) (denying institution); *Edwards Lifesciences*, IPR2023-01232, Paper 10, 36 (“Petitioner’s rationale to modify Spenser II based on inchoate and ***unasserted combinations*** is also unpersuasive.”) (denying institution).

But even if these references were to be considered, they are irrelevant. The known-technique rationale only applies when the references share the same “field of endeavor” or concern “similar devices.” *Intel*, 61 F.4th at 1379–81. Petitioner does not attempt to demonstrate that this is so. Instead, as Dr. Rubin explains, none of Orsini, Birrell, or Cochran demonstrate that it was “known” to “predefine [the] number of the lowest amount of the plurality of data chunks” in Foster’s field:

Petitioner's reliance upon Orsini, Birrell, and Cochran is misplaced as each of these references arise in an entirely different context from Foster's streaming system.

Foster concerns a streaming system. Foster's "field of the invention" "relates to a method, system and program product for attaching a title key to encrypted ***content for synchronized transmission to a recipient.***" Ex. 1006, [0001]-[0002]. Foster further clarifies that its "method, system and program" are used for "stream[ing]" "content such as video and audio data." *Id.*, [0004]. Indeed, the figure Petitioner relies upon (*e.g.*, Pet., 61, 63, 66, 69) "depicts a flow diagram of an elementary content ***stream.***"

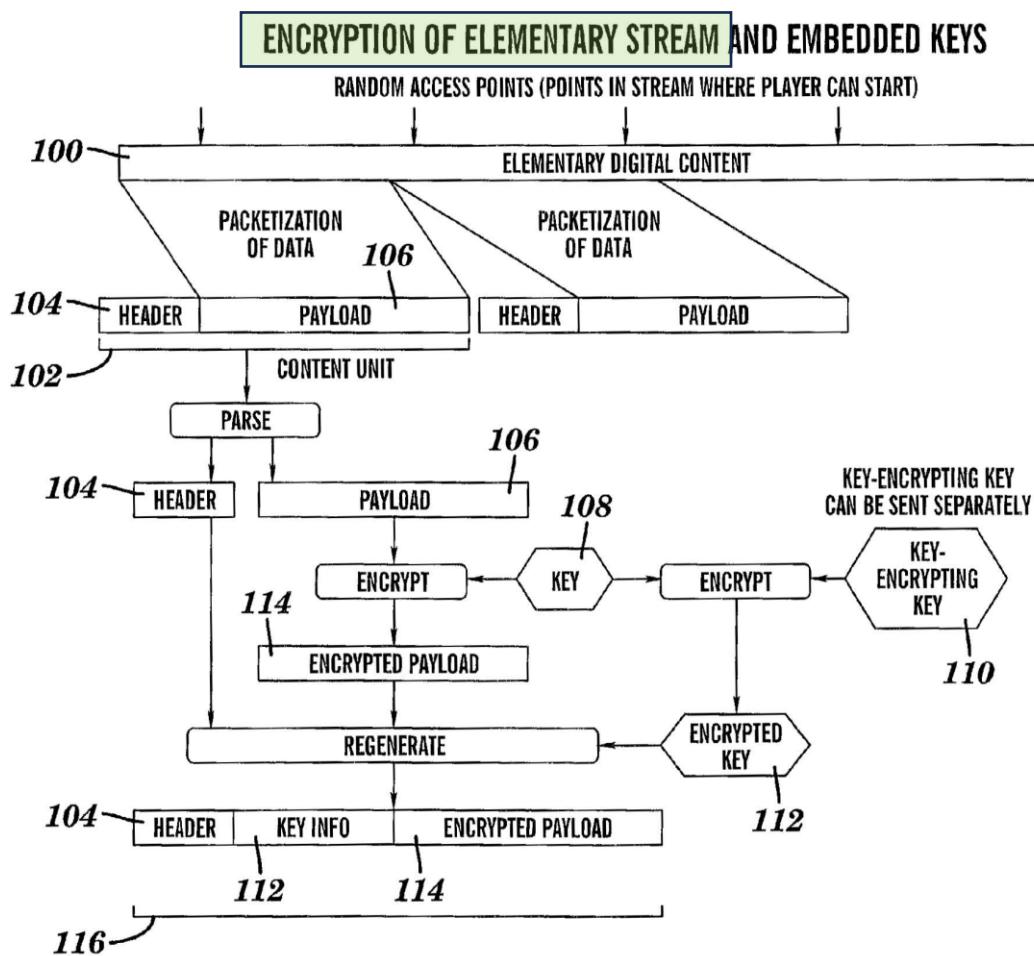


FIG. 8

Ex. 1006 [Foster] Fig. 8, [0022]; *see also* [0050] (“the content can be delivered through streaming delivery”), [0053], [0060]-[0061], [0063], [0078].

Orsini, Birrell, and Cochran, by contrast, do not generally relate to streaming. Orsini does not mention streaming and instead concerns “a system for securing data from unauthorized access or use.” Ex. 1005 [Orsini] [0003]; *id.*, [0005] (addressing need “for ensuring data stored and transmitted over [computer] systems cannot be read or otherwise

compromised.”). Birrell and Cochran similarly have nothing to do with streaming and, instead, relate to backing-up data. Birrell, titled a “method and system for ***cooperatively backing up data*** on computers in a network,” describes as its “field of [] invention” “data archiving and, specifically, . . . ***backing up data in the context of a plurality of computers*** connected via a network.” *Id.*, 1:14-17; *see also* 1:19-22, 2:31-38, 11:11-38, 15:39-16:20. Similarly, Cochran teaches “maintaining multiple copies of data” (“mirroring”) data on multiple storage arrays. Ex. 1052 [Cochran] [0004]; *see also* Ex. 1003 [Bhattacharjee-Decl.] ¶ 191 (quoting Ex. 1052 [Cochran] [0001]) (relying upon Cochran as alleged evidence that “[m]aintenance of multiple copies of data is part of the security function in data processing operations in case data is unavailable, damaged, or lost.”).

Consequently, Orsini, Birrell and Cochran are irrelevant to whether “predefin[ing the] number of the lowest amount of the plurality of data chunks” was a known technique to apply to Foster’s streaming system.

Ex. 2003 [Rubin-Decl.] ¶¶ 94-97.

Thus, Petitioner, at best, shows that storing redundant data in a minimum number of chunks to ensure the reliability of a data backup was known in secure data storage or data back-up. This does not establish that it would have been obvious to modify Foster to include this limitation under Google’s construction. *Samsung Elecs. Co., Ltd. v. Smart Mobile Techs. LLC*, IPR2022-01248, Paper 53,

20-21 (Jan. 23, 2024) (“merely establishing that multiplexing was known in other contexts is insufficient to show how or why Yegoshin should be modified.”).

Moreover, Petitioner fails to explain why a POSITA would be motivated to modify Foster’s system to “predefine [the] number of the lowest amount of the plurality of data chunks.” As Dr. Rubin explains:

Unlike systems for securing and/or backing up data stored in a system or network, Foster’s encrypted-streaming system would not benefit from “***predefine[ing]*** the] number of the lowest amount of the plurality of data chunks.” For example, one of the “transmission systems” Foster can use is the Internet. See Ex. 1006 [Foster] [0005]. One of the main protocols therein is transmission control protocol (TCP). Ex. 2017 [IBM]; see also Ex. 2022 [InfoWorld] (cable providers used TCP/IP by 1996). In TCP, the receiver returns an acknowledgement back to the host for each packet it receives. Ex. 2017 [IBM]. Under Petitioner’s mapping, a “data chunk” is one of Foster’s packets. Pet., 65. If there is no acknowledgement, the host resends the missed packet. *Id.* This acknowledgement process results in “TCP [being] capable of ensuring reliable delivery of data.” *Id.* There simply would be no need to “predefine” the number of the lowest amount of packets necessary to reconstruct the content because TCP’s acknowledgment process already ensures reliable delivery of packets. A POSITA would thus recognize that “***predefin[ing]*** the] number of the lowest amount of the plurality of data chunks” would be inappropriate

in Foster’s transmission/streaming context and result only in wasted bandwidth and processing.

Ex. 2003 [Rubin-Decl.] ¶ 99.

Petitioner’s proffered modification, thus, is simply an “add-on” solely designed to achieve the claimed limitation without providing any benefit to Foster’s system. This is precisely the sort of hindsight driven analysis the Federal Circuit has repeatedly rejected. *Ruiz*, 357 F.3d at 1275 (using claims as a roadmap is impermissible hindsight); *NTP*, 654 F.3d at 1299 (Fed. Cir. 2011) (reversing IPR obviousness rejections as “based on improper hindsight reasoning” when “adding an RF network [in the way necessary to meet the claims] would render the RF network [in the prior art] superfluous.”); *Apple*, IPR2015-00576, Paper 7, 12 (denying institution where “Petitioner . . . fails to provide a specific explanation why [the primary reference] did not already accomplish [the asserted benefits] by itself.”); *F5 Networks*, IPR2017-00124, Paper 48, 64-65 (no motivation for modification that “serve[s] no purpose”).

Thus, Petitioner fails to prove that it would be obvious to modify Foster to satisfy limitation 1[c][2] under its construction. This is reason enough to reject Petitioner’s Foster ground. “It is the petitioner that bears the burden of proof in IPRs.” *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016). Petitioner is further obliged to set forth “[h]ow the challenged claim is to be

construed” and “[h]ow the construed claim is unpatentable.” 37 C.F.R. §§ 42.104(b)(3)-(4); *Toyota Motor Corp. v. Blitzsafe Texas, LLC*, IPR2016-00422, Paper 12, 26 (July 6, 2016) (“The ‘construction’ referred to by 37 C.F.R. § 42.104(b)(3) is the construction proposed by the Petitioner, ***one that Petitioner believes is the correct construction under applicable law*** and should apply in the involved proceeding.”).

Petitioner contends that limitation 1[c][2] is satisfied under Patent Owner’s construction but that is neither here nor there. Petitioner does not propose that construction nor has it acquiesced to it. *Carefusion Corp. v. Baxter Int’l, Inc.*, IPR2016-01456, Paper 9, 7 (Feb. 6, 2017) (denying institution where petitioner “failed to indicate that it agrees with, proposes or adopts the constructions attributed to Patent Owner.”). Rather, Petitioner maintains that only its construction is consistent with the specification and that “SFI interprets the phrase to have no meaning at all.” Ex. 2018 [Google’s-Opening-Claim-Construction-Brief] 23-24. The Board cannot apply a construction that Google does not propose and that Google insists is wrong. *Samsung Elecs. v. Uniloc 2017 LLC*, IPR2020-00046, Paper 6, 10 (Apr. 1, 2020) (“We exercise our discretion and decline Petitioner’s invitation to adopt allegedly incorrect claim constructions and institute an *inter partes* review on the basis of those constructions.”); *Hologic, Inc. v. Enzo Life Sciences, Inc.*, IPR2018-00019, Paper 21, 6 (Nov. 28, 2018) (“Petitioner does

not satisfy Rule 42.104(b)(3) when, in a proceeding applying the *Phillips* claim-construction standard, it ‘expressly disagree[s] with its proffered constructions.’”).

Notably, the Office made clear in the preamble associated with the switch from a “broadest reasonable interpretation” standard to a *Phillips*-style claim construction standard, its “agree[ment] with “[s]everal comments acknowledged that harmonizing the claim construction standards **would prevent parties from taking inconsistent positions ...**” *Changes to the Claim Construction Standard for Interpreting Claims in Trial Proceedings Before the Patent Trial and Appeal Board*, 83 Fed. Reg. 51,340, 51350 (Oct. 11, 2018); *id.*, (““With a uniform application of the *Phillips* standard, patent challengers ... **will instead be required to choose a single construction that best captures the true meaning of the patent claim ...**”).

Because Petitioner fails to show that its construction is satisfied or acquiesce to Patent Owner’s construction, Petitioner has failed to demonstrate a reasonable likelihood of prevailing as to its Foster ground.

D. Petitioner Fails To Show That Foster Discloses Or Renders Obvious “Storing, . . . At Least One Data Chunk . . . On At Least One Storage Device” (Limitation 1[f]).

Limitation 1[f] requires “**storing**, in a memory coupled to the processor, **at least one data chunk** of the plurality of data chunks with data indicative of at least one of the distinct encryption keys **on at least one storage device.**” Ex. 1001

[’609] cl. 1. Even assuming, *arguendo*, that the data packets of Foster could correspond to the chunks of Claim 1, this limitation is not satisfied by Foster because Foster does not **store** its data packets on a **storage device**, as required by Claim 1. In Foster, data is streamed to a “receiver” for immediate decoding and play on a playback device such as a set top box, not for secure long-term storage. As previously explained, this distinction goes to the fundamental difference between the ’609’s inventions—for secure storage of data—and Foster’s invention—for encrypting and streaming content from a content owner to a consumer. *See* Section III.A.

Petitioner argues that Foster discloses a “memory” on a “storage device” that is also “coupled to the processor” because Foster teaches transmitting content units to a receiver. Pet., 75. Petitioner’s argument, however, fails because Foster’s “receiver” is neither a “storage device” nor used to store chunks, as those terms are used in Claim 1.

Although Petitioner never explains what a “storage device” is, Petitioner asserts that Foster’s “receiver” (also referred to a “recipient”) “is a ‘storage device’” because data packets can be “stored on [the] receiver.” Pet., 75. Petitioner’s argument thus rests upon the unsubstantiated premise that any device that has a memory is a claimed “storage device.” This is wrong, as Dr. Rubin explains:

The '609 is directed toward “securing data from unauthorized access or use.” Ex. 1001 ['609] 1:25-29. As such, an important aspect of the '609's invention is securely storing data. *Id.*, 61:35-39 (“One advantage is the security gained from moving shares of the data to different locations on one or more data depositories or storage devices, that may be in different logical, physical or geographical locations.”); *see generally id.*, 2:40-6:61 (all embodiments in summary of invention include storing data in storage facilities or storage media). The '609 gives some examples of “suitable” storage devices:

...one or more portions are stored on ***a smart card, a government Common Access Card, any suitable physical storage device (e.g., magnetic or optical disk, USB key drive, etc.)***, or any other suitable device.

Id., 67:1-4; *see also id.*, 56:54-56 (“When one or more data shares are physically removed from the system, either by using a removable device, such as a data storage device, or by placing the share under another party's control, any possibility of compromise of secured data is effectively removed.”). Thus, the '609 provides specific examples of what is meant by a “storage device,” *i.e.*, “magnetic or optical disk, USB key drive, etc.” Each of these examples of a “storage device” is one that is primarily dedicated to long-term storage and not merely a device that has a memory for transiently storing data packets as they arrive for processing and display.

Foster teaches that its receiver, by contrast, is merely a PC, DVD player or, in Petitioner's example, a consumer set-top box. Pet., 75

(“that receiver is a ‘*storage device*’ (*e.g.*, consumer set-top box)”); Ex. 1006 [Foster] [0040] (“Receiver-***a consumer device, such as a set-top box, a DVD player, etc.***, that receives content directly from a content owner, from a content service or from another receiver within a consumer home network.”); [0048] (a “PC” uses prior-art Digital Rights Management System). While a PC, DVD player, or set-top box may be capable of storing data, none of these devices are “storage devices” as that term is used in the ’609. Rather, they are used to read and immediately play content. Any “storage” of the data in short-term memory is simply to allow the data packet to be unpacked so the stream can be played. Ex. 2021 [Cambridge-Dictionary] (defining set-top box as “an electronic device that makes it possible to watch digital broadcasts on ordinary televisions”); *cf.* Ex. 2019 [Microsoft-Computer-Dictionary] 90 (“CD player *n.* ...A device that reads the information stored on a CD.”). Indeed, Foster does not even require that “receivers” be capable of storing data but simply that they “receive[] content directly from a content owner.” Ex. 1006 [Foster] [0040]. Thus, a POSITA would not understand Foster’s “receiver” as a device primarily dedicated to long-term storage, and thus would not understand Foster to teach “storing” data chunks on a “storage device.”

Ex. 2003 [Rubin-Decl.] ¶¶ 104-105.

Thus, a POSITA reviewing the ’609 patent would not understand a “storage device” to encompass something like Foster’s receiver (*e.g.*, a set-top box or a DVD player) because the examples of storage devices given in the ’609—

“magnetic or optical disk, USB key drive, etc.”—are each devices primarily dedicated to long-term storage. While the ’609 does use the word “etc.,” that just refers to things of a similar type. *IBM v. Iancu*, 759 Fed. App’x. 1002, 1007–08 (Fed. Cir. 2019) (the word “etc.” “indicate[s] that only things of a type similar to the itemized ones are covered . . . [w]e have recognized that . . . ‘etc.’ sometimes have just that meaning.”) (citing *Archer Daniels Midland Co. v. United States*, 561 F.3d 1308, 1313 (Fed. Cir. 2009) (holding that the “rule of ejusdem generis . . . limits the additional [things] included by the general phrase ‘etc.’ to others of the types listed”)). Consequently, Petitioner fails to show that Foster discloses a “storage device.”

Tacitly admitting that Foster does not disclose “storing . . . at least one data chunk . . . on a storage device,” Petitioner “alternatively” argues that a POSITA “had reason” to modify Foster such that “Foster’s encrypted content [is stored] in the memory of storage devices” for long-term, storage. Pet., 76. Petitioner argues that a POSITA would have modified Foster for long-term storage in order to avoid “repeat[ing] the encryption processing every time digital content (*e.g.*, a movie) needed to be encrypted and delivered to recipients” and thereby allowing Foster’s system to facilitate “(1) backup, (2) video-on-demand distribution, and/or (3) distribution to later-identified recipients.” *Id.*, 77. Petitioner further contends that so storing the content would have been applying a “known technique” to solve a

known problem of “how to store encrypted content for delivery to multiple recipients at varying times . . .” *Id.*

But, Petitioner’s proposed modification does not actually solve any problem. As Dr. Rubin explains, a POSITA would recognize that the purported benefits Petitioner identifies vis-à-vis “(1) backup, (2) video-on-demand distribution, and/or (3) distribution to later-identified recipients,” are already solved by Foster’s system because those benefits are provided by the content owner:

Foster’s system is meant merely to act as a middleman for encryption and transmission of content (*e.g.*, a movie) from **content sources** (*e.g.*, a movie studio) to **consumer recipients**, as can be seen in Figure 16 below.

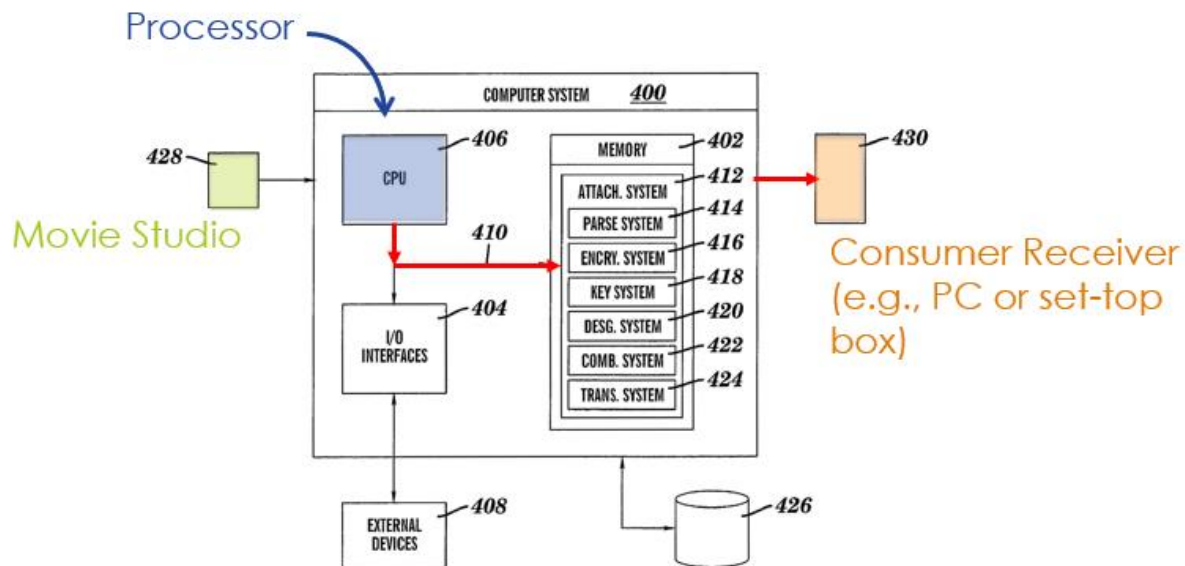


FIG. 16

Ex. 1006 [Foster] Fig. 16 (annotated); [0001] (“a method, system and program product for attaching a title key to encrypted content for synchronized transmission to a recipient.”); [0078] (“computer system 400 is intended to be representative of any system capable of processing content units of an elementary content stream received from source 428, and transmitting the processed content units to a recipient 430.”).

Foster further teaches that the content “source” or “owner” is an entity such as a movie studio that owns and provides the content. *Id.*, [0038], [0042]. A POSITA would thus recognize that storage for “(1) backup, (2) video-on-demand distribution, and/or (3) distribution to later-identified recipients” is already provided by the content source/owner. Indeed, none of the references Petitioner cites as alleged background knowledge of a POSITA teaches long-term storage of content by an intermediary or at a receiver, as in Foster’s system. See Exs. 1045, 1050, 1060.

Ex. 2003 [Rubin-Decl.] ¶¶ 107-108.

Moreover, Petitioner’s proposed modifications would either compromise the streamed content’s security or require a significant amount of resources to prevent such a thing, as Dr. Rubin explains:

The entire premise of Foster’s system is that it can receive content ***without*** storing it, encrypt the content, and distribute it to consumers. See Ex. 1006 [Foster] [0001]. Such a system entails numerous benefits for it ensures the content is secure. By contrast, if Foster’s system were altered so that the content were stored in Foster’s

system, that would create a significant security risk as Foster's system would be hacked and an adversary would only have to target the key management block on a single consumer receiver to compromise all of the stored pre-encrypted content. There is simply no reason why a POSITA would be motivated to do this as it would surely frustrate Foster's purpose of securing the content (vis-à-vis encryption) and transmitting it to the receiver without exposing the content to further risk. As discussed, a POSITA would recognize that content is already securely stored in Foster's content sources/owners. A POSITA—and, more importantly, the content owners—would not be motivated to expose the content to potential theft unnecessarily. Moreover, if Foster's system were modified to store pre-encrypted content, significantly more storage would have to be added and that additional storage would have to be as secure as the storage at the content provider/owner. Otherwise, content provider/owners would never trust a third-party system to store its content. Indeed, Foster teaches security of the content is of such a concern that the title keys used to encrypt individual content packets need to be changed relatively "frequently." Ex. 1006 [Foster] [0055].

Ex. 2003 [Rubin-Decl.] ¶ 110. Thus, Petitioner's proposed modification is not needed by Foster and would, in fact, frustrate Foster's purpose by comprising security. A POSITA would not be motivated to make Foster's system worse.

Polaris Indus., Inc. v. Arctic Cat, Inc., 882 F.3d 1056, 1069-70 (Fed. Cir. 2018) (reversible error to find motivation to combine where resulting combination

“would run contrary to [the primary reference’s] stated purposes.”); *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 [POSITA] to pursue a combination with that reference.”); *In re Ratti*, 270 F.2d 810, 813 (CCPA 1959) (no motivation to combine where “suggested combination of references would require . . . a change in the basic principles under which the [reference] was designed to operate.”); *Arctic Cat Inc. v. Polaris Indus.*, 795 F. App’x. at 833 (“The 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*, 938 F.3d at 1331-32 (rejecting argument “Board placed undue weight on the disadvantages of introducing Iwaguchi’s TPM sensor into Kauffman’s system” where tradeoffs of combination may “yield an unappetizing combination”).

E. Petitioner Fails To Show That Foster Teaches The Claimed Steps In Order.

Petitioner contends that the steps in claim 1 “must be performed in order.”² Pet., 78 (“Google’s construction [is] that claim 1’s steps be performed in order.”);

² To be sure, Patent Owner has not adopted this position and reserves the right to take a contrary position in District Court.

Ex. 1064 [Google-Proposed-Claim-Construction] 16. Consequently, per Petitioner, “receiving a first key from a storage system” must be the first step performed in Foster’s method relative to the other steps required by the claim. Ex. 1001 [’609] cl. 1.

Petitioner admits that Foster does not disclose “receiving a first key” as the first step in the claim, admitting that “Foster does not limit when it receives a KEK.” Pet., 78. Thus, Petitioner’s argument lies in obviousness. Petitioner argues that, while Foster does not expressly limit when it receives a KEK, it renders the limitation obvious because (1) it would have been a conventional design choice to receive the KEK as a first step in Foster’s method, or (2) when encrypting several data sets back-to-back with the same KEK, POSAs would have understood that the system has already received the KEK from encrypting a previous data set, before it encrypts the next data set. *Id.*

Petitioner does not explain why “receiv[ing] the KEK as a first step” “would have been a conventional design choice.” Nor does Petitioner explain how Foster teaches that “several data sets” would have been encrypted “back-to-back with the same KEK.” These conclusory assertions fail as a matter of law to establish obviousness. *KSR*, 550 U.S. at 418 (citing *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)) (“[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with

some rational underpinning to support the legal conclusion of obviousness.”); *Magnum Oil*, 829 F.3d at 1380 (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements. The petitioner must instead articulate specific reasoning, based on evidence of record, to support the legal conclusion of obviousness.”).

Petitioner cites no evidence for its assertions except Dr. Bhattacharjee’s equally unsupported declaration which just repeats the Petition’s conclusory argument. Ex. 1003 [Bhattacharjee-Decl.] ¶ 214 (parroting Petition). It is well-settled that such conclusory arguments should be afforded little weight by the Board. *See, e.g., Xerox*, IPR2022-00624, Paper 9, 15; 37 C.F.R. § 42.65(a); *Ashland Oil, Inc. v. Delta Resins & Refractories, Inc.*, 776 F.2d 281, 294 (Fed. Cir. 1985) (“Lack of factual support for expert opinion going to factual determinations . . . may render the testimony of little probative value in a validity determination.”).

Moreover, as Dr. Rubin explains, it would not be a “conventional design choice” to modify Foster’s system to “receive the KEK as a first step” nor does Foster teach “encrypting several data sets back-to-back with the same KEK”:

Foster suggests precisely the opposite of “receiving the KEK as a first step”, suggesting that the **KEK is received *after*** the content is “**packetiz[ed]**” and “**encrypt[ed]**” rather than before:

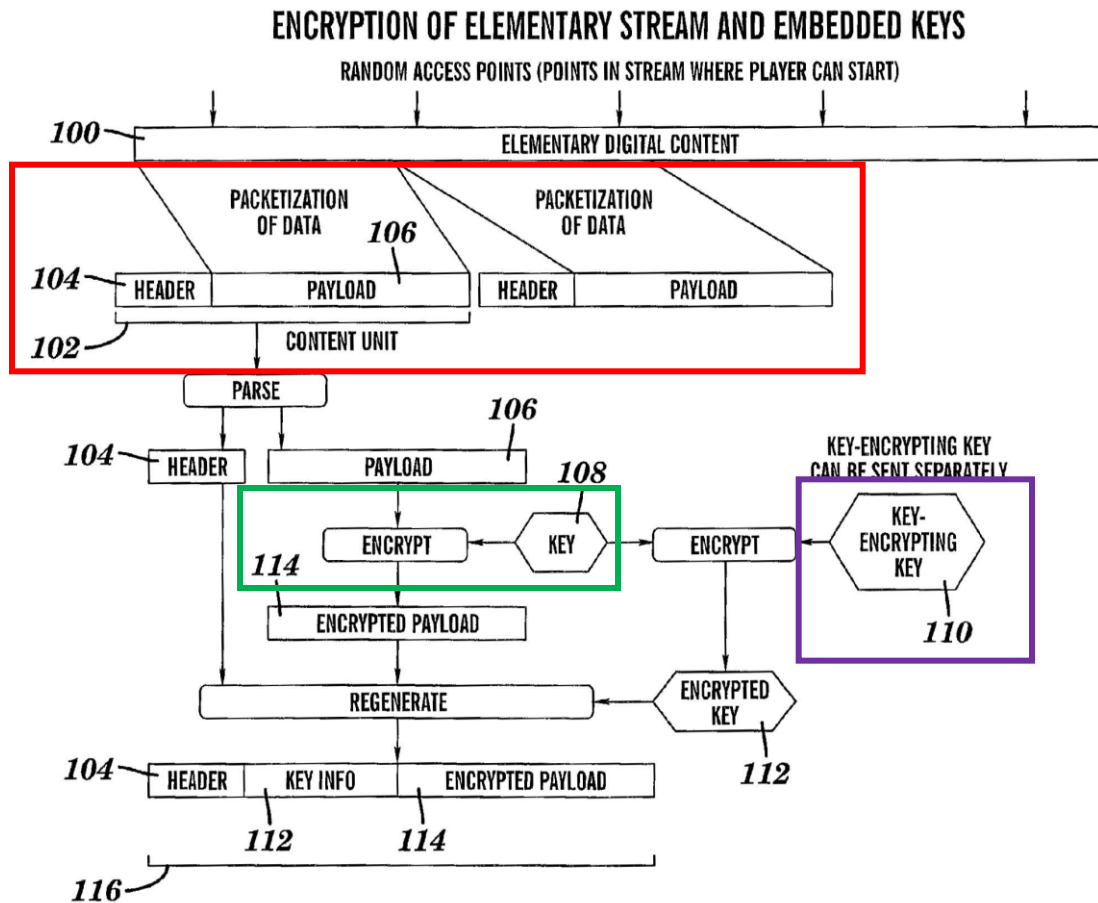


FIG. 8

Ex. 1006 [Foster] Fig. 8 (annotated).

Moreover, any design choice concerning keys—including when they are received and whether the system keeps them between different encryption sessions—cannot be merely “conventional.” Rather, these choices must be carefully considered and the threat model properly assessed because, as previously explained, key management is the most difficult and most vulnerable part of cryptography. See Ex. 2010 [Schneier-Applied-Cryptography] 169 (“In the real world, key management is the hardest part of cryptography. Designing secure

cryptographic algorithms and protocols isn't easy, but you can rely on a large body of academic research. Keeping the keys secret is much harder.”). In accord, Petitioner acknowledges that a POSITA would have been concerned about key security, especially internal to the cryptographic engine. Pet., 23-26. Indeed, a reason to use a key just for one communications session is because doing so eliminates the need for internal storage and its associated risks. Ex. 2010 [Schneier-Applied-Cryptography] 180 (“There is no reason to store the key after it has been used. And if you use some key-exchange protocol to transfer the key from one conversant to the other, the key doesn't have to be stored before it is used either. This makes it far less likely that the key might be compromised.”). Consequently, a POSITA would not have considered when to receive the first key or whether to keep the first key in between encryption sessions to be a “conventional design choice.”

Ex. 2003 [Rubin-Decl.] ¶¶ 116-117.

Thus, Petitioner fails to demonstrate that Foster renders obvious “receiving a first key from a storage system” performed as the first step.

IV. CONCLUSION

Thus, 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. Elliott /

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: February 28, 2024

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITS

This Patent Owner Preliminary Response (the “POPR”) consists of 13,111 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,

/ Colette Woo /

Date: February 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 2003-2022

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

Elisabeth H. Hunt	EHunt-PTAB@wolfgreenfield.com
Gregory S. Nieberg	GNieberg-PTAB@wolfgreenfield.com
Nathan R. Speed	nspeed@wolfgreelfield.com

Respectfully submitted,

/ Colette Woo /

Date: February 28, 2024